27th BRAZILIAN SYMPOSIUM ON DATABASES

9th Demos and Applications Session

PROCEEDINGS

October 15th-18th, 2012
São Paulo, São Paulo, Brazil

Promotion

Brazilian Computer Society - SBC
SBC Special Interest Group on Databases

Organization

Universidade de São Paulo - USP

Realization

Instituto de Matemática e Estatística - USP

Demos and Applications Session Chairs

José Maria da Silva Monteiro Filho (UFC)
Javam de Castro Machado (UFC)

Editorial

The Brazilian Computer Society (SBC) promotes annually the Brazilian Symposium on Databases (Simpósio Brasileiro de Banco de Dados – SBBD, in Portuguese). It has been the largest venue in Latin America for presenting and discussing research results in the database domain as well in other related areas such as information retrieval, digital libraries, knowledge discovery and data mining. In its 27th edition, the symposium will be held in the city of São Paulo, in the state of São Paulo, on October 15-18, 2012. SBBD aggregates researchers, students and practitioners, from Brazil and abroad, for discussing problems and research results related to the main topics in modern database technologies.

During SBBD, since 2004, a special session is dedicated to the presentation of database software aggregated with research solutions. This session, named Demonstrations Session (or Demos Session, for short), aims at divulging practical achievements among researchers, developers and professionals, from both academia and industry, interested in relevant and functional tools that use data/information management technologies to solve non-trivial problems. The demonstrations held in the Demos Session concern relevant problems related to data management technologies, providing interaction between researchers and practitioners.

In this edition's issue, we had 13 demos submissions, each paper was evaluated by 3 reviewers selected from a committee of 22 researchers from different Brazilian institutions. At the end of the reviewing process, the program committee selected 6 papers to be presented and demonstrated in the Demos Session. During the presentations of the demos they are assessed again by a committee, which will select the best demo. We thank all authors of submitted papers for their interest in the event. We also thank the reviewers committee for its prompt and careful evaluations. We would like to express our gratitude for their hard work, which was fundamental for this event to happen.

We also congratulate the SBBD 2012 organizers for the local arrangements that provide the necessary infrastructure for Demos exposition, as well as to SBBD Steering for the continuous support given to SBBD Demos, including the eventual financial aid for students of accepted Demos to attend the symposium.

We hope the SBBD 2012 Demos Session can awake great insights and many interaction opportunities for the Brazilian Database community.

José Maria da Silva Monteiro Filho, DC-UFC

Javam de Castro Machado, DC-UFC

SBBD 2012, Demos and Applications Session Chairs

27th BRAZILIAN SYMPOSIUM ON DATABASES

October 15th – 18th, 2012

São Paulo, São Paulo, Brazil

Promotion

Brazilian Computer Society - SBC
SBC Special Interest Group on Databases

Organization

Universidade de São Paulo - USP

Realization

Instituto de Matemática e Estatística - USP

SBBD Steering Committee

José Palazzo Moreira de Oliveira (UFRGS) – Chair
Angelo Brayner (UNIFOR)
Alberto Laender (UFMG)
Cláudia Bauzer Medeiros (UNICAMP)
Cristina Dutra de Aguiar Ciferri (ICMC-USP)
Marco A. Casanova (PUC-Rio)

SBBD 2012 Committee

Steering Committee Chair

José Palazzo Moreira de Oliveira (UFRGS)

Local Organization Chair

João Eduardo Ferreira (IME-USP)

Program Committee Chair

Marco A. Casanova (PUC-Rio)

Short Papers Chair

Renata Galante (UFRGS)

Demos and Applications Chairs

José Maria da Silva Monteiro Filho (UFC)
Javam de Castro Machado (UFC)

Thesis and Dissertation Workshop Chairs

Fabio Porto (LNCC)
Ana Maria de C. Moura (LNCC)

Tutorials Chair

Cristina Dutra de Aguiar Ciferri (ICMC-USP)

Lectures Chair

Marcio K. Oikawa (UFABC)

Local Organization Committee

João Eduardo Ferreira (IME-USP) – Chair
Isabel Italiano (EACH-USP)
Kelly R. Braghetto (UFABC)
Luciano V. Araújo (EACH-USP)
Marcio K. Oikawa (UFABC)

Demos and Applications Program Committee

Ana Carolina Salgado (UFPE)
André Santanchè (Unicamp)
Angelo Roncalli Alencar Brayner (UNIFOR)
Agma Traina (ICMC – USP)
Altigran Soares da Silva (UFAM)
Bernadette Farias Loscio (UFPE)
Caetano Traina (ICMC – USP)
Carmem Satie Hara (UFPR)
Daniela Barreiro Claro (UFBA)
Edleno de Moura (UFAM)
Elaine Sousa (ICMC-USP)
Fernanda Araujo Baiao (UNIRIO)
Fernando da Fonseca de Souza (UFPE)
Geraldo Zimbrão (UFRJ)
Humberto Luiz Razente (UFABC)
Jonice Oliveira (UFRJ)
José Antônio F. de Macêdo (UFC)
Karin Becker (UFRGS)
Luciano A. Digiampietri (USP)
Marcela Xavier Ribeiro (UFSCar)
Márcio Oikawa (UFABC)
Maria Camila Nardini Barioni (UFABC)
Marta Mattoso (COPPE – UFRJ)
Mirella M. Moro (UFMG)
Renata Galante (UFRGS)
Renato Bueno (UFSCar)
Renato Fileto (UFSC)
Ricardo Torres (Unicamp)
Ronaldo dos Santos Mello (UFSC)
Sahudy Montenegro González (UFSCar Sorocaba)
Sandra de Amo (UFU)
Valeria Cesario Times (UFPE)
Vanessa Braganholo (UFF)
Vânia Maria Ponte Vidal (UFC)

27th BRAZILIAN SYMPOSIUM ON DATABASES

Demos and Applications Session

Table of Contents

CPrefSQL-Tool: Uma Ferramenta Web para Consultas com Suporte a Contextos e Preferências do Usuário

Vinicius V. S. Dias[1], **Sandra de Amo**[1]

[1] Faculdade de Computação
Universidade Federal de Uberlândia (UFU) – Uberlândia, MG – Brazil

viniciusvdias@comp.ufu.br,deamo@ufu.br

Abstract. *In an ubiquitous computing scenario users access to databases will not occur at a unique context, and consequently their expectation when querying data will vary according to a multitude of parameters including location, time and surrounding influences. In this article we present CPrefSQL-Tool, a web tool allowing to execute queries and filter the results according to the user preferences and contexts. Some of the functionalities of the CPrefSQL-Tool are: (1) queries may be specified either by using standard SQL commands or by using CPrefSQL, an extension of SQL with specific operators for computing the* $top-k$ *and the most preferred tuples according to a set of contextual preference rules; (2) preference rules are easily specified by means of a simple and intuitive interface; (3) queries are executed by using the CPrefSQL query processor; (4) the filtered results of a query execution are traceable and explained on user demand.*

Resumo. *Num cenário típico de computação ubíqua, os acessos dos usuários a um banco de dados não ocorrem em um único contexto e consequentemente suas expectativas ao consultar os dados podem variar de acordo com uma multitude de parâmetros incluindo sua localização, o momento presente e as influências do ambiente. Este artigo apresenta CPrefSQL-Tool, uma ferramenta Web que permite a execução de consultas e a filtragem dos resultados levando em consideração as preferências e o contexto do usuário. Algumas das funcionalidades de CPrefSQL-Tool são: (1) as consultas podem ser especificadas por meio de comandos SQL ou por meio de comandos da linguagem CPrefSQL, uma extensão de SQL com operadores específicos para computar as* k *melhores tuplas de acordo com um conjunto de regras de preferências contextuais fornecido; (2) as regras de preferência são facilmente especificadas através de uma interface simples e intuitiva; (3) as consultas são executadas por meio do processador de consultas da linguagem CPrefSQL; (4) os resultados filtrados após a execução de uma consulta podem ser rastreados e explicados caso o usuário assim o desejar.*

1. Introdução

A incorporação de recursos permitindo levar em conta as preferências e o contexto dos usuários em mecanismos de busca ou em linguagens tradicionais de consultas em banco de dados tem despertado cada vez mais interesse na comunidade científica, abrindo possibilidades de pesquisa em áreas como banco de dados, inteligência artificial e recuperação de informação. Isso pode ser observado em aplicações sensíveis a contexto, como definições de perfis de usuário na *Web* [Suryanarayana and Hjelm 2002]

e computação móvel [Chen and Kotz 2000], que garantem uma experiência personalizada para o usuário. No que diz respeito à área de banco de dados, a pesquisa tem se concentrado na incorporação de operadores de preferências às linguagens de consultas tradicionais para manipulação de dados, como o SQL [de Amo and Pereira 2010, K.Stefanidis and E.Pitoura 2008]. Recentemente, foi proposta a linguagem CPrefSQL [de Amo and Pereira 2010], que estende o SQL padrão com dois operadores que permitem filtrar todas as melhores tuplas ou as k melhores tuplas de acordo com uma hierarquia de *preferências contextuais* informada pelo usuário através de comandos específicos da linguagem. O modelo de preferências adotado em CPrefSQL adapta o formalismo lógico de [Wilson 2009].

Neste artigo é apresentada *CPrefSQL-Tool*, uma ferramenta *Web* desenvolvida com o propósito de possibilitar que usuários, entendidos ou não da linguagem CPrefSQL, possam gerenciar e aplicar suas preferências num banco de dados de sua escolha. O foco principal deste trabalho está em disponibilizar as funcionalidades da linguagem CPrefSQL através de uma interface simples e intuitiva. A ferramenta também permite ao usuário entender como o processador de CPrefSQL executa a filtragem das tuplas de acordo com as regras de preferências contextuais fornecidas pelo usuário. Mais ainda, a ferramenta permite comparar os resultados retornados por consultas tradicionais (SQL padrão) e consultas CPrefSQL. Está disponível em http://www.lsi.ufu.br/cprefsql.

2. A Linguagem CPrefSQL

A linguagem CPrefSQL possui três funcionalidades básicas, que envolvem operações de (1) criação de preferências contextuais, (2) consulta de tuplas mais preferidas e (3) *ranking* de tuplas preferidas. A seguir, será apresentada cada uma das funcionalidades da linguagem juntamente com seu comando correspondente. Considere, para os exemplos ao longo deste texto, um banco de dados composto pela relação *hospedagem(hotel, cidade, avaliacao, preco, distancia, finalidade)*. Cada tupla desta relação descreve uma opção de hospedagem, com informações de nome de hotel, localização, preço da diária, distância e finalidade da mesma respectivamente.

2.1. Criação de Preferências Contextuais

No modelo considerado, definimos uma preferência de usuário como um conjunto de regras do tipo *"if <contexto> then <preferencia>"*, onde *<preferencia>* da regra indica a vontade do usuário em uma situação que satisfaz o *<contexto>*. Dito isso, a criação de uma preferência consiste em executar um comando CPrefSQL que recebe um conjunto de regras deste tipo e as armazena no catálogo do SGBDR (Sistema Gerenciador de Banco de Dados Relacional) através de um identificador. A Figura 1(a) ilustra o comando de criação de preferências contextuais. O comando da Figura 1(b) ilustra a criação de uma preferência sobre a relação *hospedagem*. Basicamente ele cria uma preferência com o nome "mypref-hospedagem"que envolve atributos da relação *hospedagem*. Por exemplo, a primeira regra pode ser traduzida como:"*Se a distância for menor do que 500km, então prefiro hospedagens com diária inferior a R$ 250,00 desde que as finalidades respectivas das mesmas sejam idênticas*.

```
CREATE PREFERENCES
<nome-pref>
FROM <tabelas> AS
<regras-pref>
```
(a)

```
CREATE PREFERENCES mypref-hospedagem
FROM hospedagem AS
IF distancia<500 THEN preco< 250 > preco≥ 250 [1,2,3] AND
IF preco>400 THEN avaliacao=5 > avaliacao=4 [1,2] AND
finalidade = 'férias' > finalidade = 'trabalho' [1,2,5]
```
(b)

Figura 1. Sintaxe do Comando de Criação de Preferências

2.2. Tuplas mais Preferidas e *Ranking* de Tuplas

Após a criação de uma preferência, a mesma pode ser usada em alguma consulta de CPrefSQL. Para entender como as preferências são usadas pelo processador CPrefSQL, é necessário discutir brevemente a semântica de um conjunto de regras de preferências. Um tal conjunto induz uma ordem natural sobre as tuplas de um banco de dados. Esta ordem constitui sua *semântica*. Cada regra de preferência isolada induz uma relação de dominância entre as tuplas de forma natural. Por exemplo, considere as tuplas t_1 = (Excelsior, Madri, 3, 190, 400, férias) e t_2 = (St Germain, Paris, 5, 300, 400, férias). A primeira regra "**IF** distancia<500 **THEN** (preco $<$ 250) $>$ (preco $\geq$ 250) [1,2,3]" permite comparar t_1 e t_2 e concluir que t_1 é preferida a t_2 (ou t_1 *domina* t_2). Repare que duas tuplas podem ser comparadas se elas indiferem nos atributos que não aparecem na regra (finalidade - atributo 6). Os números entre colchetes [1,2,3] referem-se respectivamente aos atributos *hotel, cidade* e *avaliação* que não precisam ser idênticos nas duas tuplas a fim de que estas possam ser comparadas. A relação de ordem final (ou relação de *dominância* entre as tuplas) correspondente a um *conjunto* de regras consiste em considerar a união das ordens induzidas por cada regra individualmente e depois considerar o fecho transitivo da relação obtida.

A relação de dominância pode ser visualizada através de um *Grafo de Dominância*. Dado este grafo, diz-se que uma tupla t_1 domina uma tupla t_2 se existir um caminho direcionado que parta de t_1 e chegue em t_2. A ausência de arcos entre tuplas indica tuplas *indiferentes* e arcos tracejados representam uma dominância por transitividade. Assim, no exemplo da Figura 2, temos que: t_1 é *dominante* em relação à t_2 e t_3, t_2 é *dominante* em relação à t_3 mas *dominada* em relação à t_1.

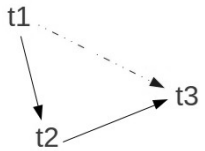


Figura 2. Grafo de Dominância

Deste modo, a ideia dos algoritmos que implementam a linguagem CPrefSQL é, justamente, percorrer todas as tuplas em questão e calcular a dominância entre as mesmas, que é definida para cada par de tuplas consideradas. No final resta a exibição das tuplas escolhidas para a resposta de acordo com a operação solicitada (mais preferidas ou *ranking*).

Dois tipos de consultas podem ser feitas: (1) perguntar pelas tuplas que melhor se adequam a uma dada preferência de usuário; (2) pedir as $top-k$ tuplas, isto é as primeiras k tuplas preferidas de acordo com o *ranking* estabelecido pelas regras de preferências criadas. No primeiro tipo de consulta, as dominâncias entre as tuplas são calculadas e as consideradas *dominadas* (possuem pelo menos um arco de entrada de acordo com o *Grafo de Dominância* correspondente) são excluídas da resposta. Aqui, a ordem de apresentação das tuplas do resultado não tem significado, a única certeza sobre elas é que são preferidas (*dominantes*). No segundo tipo de consulta, as tuplas da resposta são ordenadas em ordem decrescente de preferência e as k primeiras são retornadas. Essa ordenação é feita do seguinte modo: (1) a cada tupla é associado um nível, (2) de acordo com o *Grafo de Dominância*, o processo é iniciado atribuindo-se o nível 0 às tuplas que não são dominadas por nenhuma outra tupla (as que não possuem nenhum arco de entrada), (3) para as

tuplas restantes, os seus respectivos níveis são calculados pelo maior dos níveis de seus pais (dominantes) acrescido de uma unidade. Ao final, as k tuplas com o menor nível entram na resposta da consulta e entre as de mesmo nível o critério é pela sua ordem no banco de dados. Seguindo esse raciocínio, para o grafo da Figura 2: t_1 tem nível 0, t_2 tem nível 1 (max(0) + 1) e t_3 tem nível 2 (max(0,1) + 1). Portanto, pode-se organizar as tuplas na seguinte ordem decrescente de preferência: t_1 , t_2 e t_3 .

A Figura 3(a) ilustra o código do bloco padrão de uma consulta SQL fornecendo as k tuplas preferidas. Caso se deseje todas as tuplas preferidas (não dominadas), o parâmetro k é instanciado como -1. O comando da Figura 3(b) recupera as 3 tuplas mais preferidas da relação *hospedagem* de acordo com as regras de preferência *mypref_hospedagem* especificadas na Figura 1(b).

```
SELECT <lista-atributos>
FROM <tabelas>
[WHERE <restricoes-where>]
ACCORDING TO PREFERENCES (<nome-pref>, < k >)
[GROUP BY <lista-atributos>]
[ORDER BY <lista-atributos>]

* Os comandos apresentados entre colchetes são
opcionais.
```

(a)

```
SELECT *
FROM hospedagem
ACCORDING TO PREFERENCES
(mypref-hospedagem, 3);
```

(b)

Figura 3. Bloco padrão de CPrefSQL para as *k* tuplas mais preferidas

3. As Funcionalidades da Ferramenta CPrefSQL-Tool

Gerenciamento de Banco de Dados: Todo o gerenciamento do banco de dados padrão que não envolve preferências pode ser realizado dentro da própria ferramenta. **Gerenciamento de Preferências:** Permite criar, alterar, e excluir regras de preferências contextuais. Estas operações podem ser realizadas através de comandos CPrefSQL ou através de uma interface de mais alto nível. Aqui, o usuário poderá entrar suas regras de preferências contextuais. O conjunto de regras de preferências é criado pelo sistema e um identificador é associado ao mesmo. Através deste identificador, as preferências podem ser alteradas e/ou excluídas futuramente. A interface permite que isso seja feito sem a necessidade de se conhecer a sintaxe da linguagem CPrefSQL, ou seja, o usuário pode criar regras contextuais, associá-las a uma preferência e fazer consultas sobre a mesma sem a necessidade de um conhecimento específico dos comandos apresentados nas Figuras 1 e 3. A Figura 4 ilustra a interface de inserção das regras de preferências pelo usuário, correspondente ao comando de criação de preferências apresentado na Figura 1(b). **Configurações:** As configurações permitem ao usuário a portabilidade entre ambientes e a customização do acesso à ferramenta, como disponibilizado pela maioria dos SGBDs. São elas: **(A) Acesso –** bancos de dados *locais* ou *remotos*; **(B) Diretório de preferências –** Todo o armazenamento, exclusão e edição de preferências será feito neste diretório (padrão: */var/lib/pgsql/*); **(C) Visualização do Resultado –** resultado da consulta CPrefSQL com ou sem o resultado da consulta realizada com SQL padrão, permitindo a comparação entre dois cenários (com e sem preferências). **(D) Rastreamento de Resultado –** relaciona as regras de preferência com a resposta dada de uma consulta submetida através de um registro que mostra as relações de dominância entre as tuplas, dando uma noção do funcionamento dos algoritmos da linguagem CPrefSQL; **(E) Construção de Consultas –** submissão de consultas usando a sintaxe de CPrefSQL ou alternativamente usando a sintaxe padrão de SQL incluindo a escolha de um conjunto de regras de preferências através

de seu nome (ou identificador). Tais regras já devem ter sido criadas através do comando CPrefSQL (ver Figura 1(a)) ou usando a interface da ferramenta própria para isso.

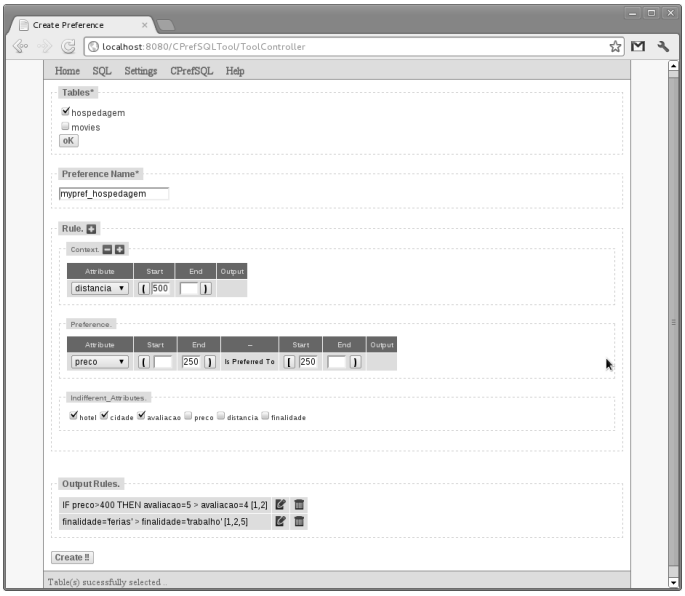


Figura 4. Interface de Inserção das Regras de Preferências

Execução de Consultas: De acordo com a configuração feita em (E), o usuário (1) insere a consulta SQL e o nome do arquivo de preferências ou (2) insere o comando CPrefSQL correspondente à sua consulta (caso conheça seja familiarizado com a sintaxe desta linguagem). A Figura 5 ilustra a execução da consulta da Figura 3(b), em que as seguintes configurações foram declaradas: (C) apresentar apenas o resultado da consulta CPrefSQL, (D) rastreamento de resultado desativado e (E) definido como SQL padrão mais a escolha do nome de um conjunto de regras de preferência criado.

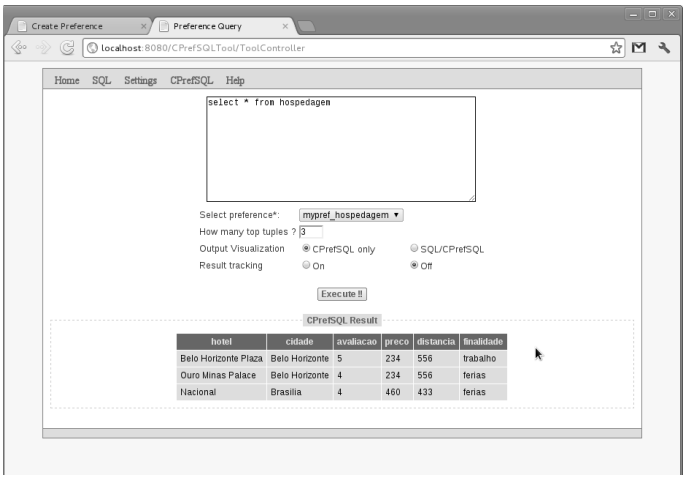


Figura 5. Execução de uma consulta

4. Implementação e Ambiente

A ferramenta foi implementada como uma aplicação *Java-Web* por motivos de portabilidade. Assim, a aplicação oferece a interface para manipulação de preferência sobre bancos de dados armazenados em um SGBDR, local ou remoto. Além disso, devido às características originalmente propostas para a linguagem CPrefSQL e suas extensões, a ferramenta usa o PostgreSQL como SGBDR. A arquitetura simplificada da ferramenta pode ser observada na Figura 6. Assim: (1) um usuário interage com o servidor *Web* da

ferramenta através de uma interface HTTP Requisição/Resposta, (2) o servidor se comunica com o banco de dados através de consultas (SQL ou CPrefSQL) e obtém respostas do mesmo, (3) no caso de a requisição ao SGBDR envolver preferências CPrefSQL, é necessário que o processador de consultas se comunique com o diretório de preferências.

Outro ponto importante é que o gerenciador de banco de dados a ser usado deve suportar a linguagem CPrefSQL, demandando uma pré-configuração desse ambiente.

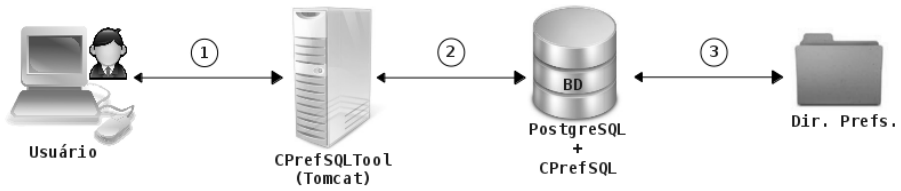


Figura 6. Arquitetura da ferramenta CPrefSQLTool

5. Conclusões e Extensões Futuras

Neste artigo foi apresentada *CPrefSQL-Tool*, uma ferramenta *Web* que permite a execução de consultas com suporte a preferências e contextos do usuário de forma simples e amigável. Com estas características, *CPrefSQL-Tool* é capaz, através do exercício prático, de estabelecer fortes vantagens em se utilizar linguagens com suporte a preferências como a CPrefSQL. Neste aspecto, o objetivo é que um usuário possa criar preferências sobre um banco de dados qualquer e, através de consultas envolvendo as mesmas, perceber as diferenças na resposta obtida quando comparada a uma consulta SQL padrão. Diferenças que, em geral, fazem com que a partir de uma linguagem com suporte a preferências os piores casos (consultas sem resposta ou com respostas inviavelmente volumosas) sejam evitados. Um trabalho futuro é a incorporação de um módulo de mineração de preferências [de Amo et al. 2012, da Silva and de Amo. 2011] na ferramenta, possibilitando uma funcionalidade de extração automática de preferências do usuário a partir de suas escolhas passadas.

Referências

Chen, G. and Kotz, D. (2000). A survey of context-aware mobile computing research. *Technical report tr2000, Departament of Computer Science, Dartmouth College, Hanover, New Hampshire, USA*, page 381.

da Silva, N. F. F. and de Amo., S. (2011). Cprefminer: A bayesian miner of conditional preferences. *Journal of Information and Data Management (JIDM)*, 2 (1):35–42.

de Amo, S., Diallo, M. S., Diop, C. T., Giacometti, A., Li, H. D., and Soulet, A. (2012). Mining contextual preference rules for building user profiles. In *14th International Conference on Data Warehousing and Knowledge Discovery - DaWaK 2012, Vienna (Austria)*.

de Amo, S. and Pereira, F. (2010). Evaluation of conditional preference queries. *Journal of Information and Data Management (JIDM).*, 1(3):521–536.

K.Stefanidis and E.Pitoura (2008). Fast contextual preference scoring of database tuples. In *Proceedings of the International Conference on Extending Database Technology (EDBT)*, pages 344–355.

Suryanarayana, L. and Hjelm, J. (2002). Profiles for the situated web. *11th International World Wide Web Conference (WWW 2002)*, pages 200–209.

Wilson, N. (2009). Extending cp-nets with stronger conditional preference statements. *AAAI*.

SAHA: sistema para acompanhamento holístico de atletas

Frederico C. da Silva[1], Fabio Porto[1], Ana Maria de C. Moura[1], Daniele C. Palazzi[1], Luis Eduardo Viveiros de Castro[2,3], Adriana Bassini[3], L. C. Cameron[3]

[1]Extreme Data Laboratory (DEXL Lab)
Laboratório Nacional de Computação Científica (LNCC)
Petrópolis – RJ – Brasil

[2]Comitê Olímpico Brasileiro - COB
Rio de Janeiro – RJ – Brasil

[3]Laboratório de Bioquímica de Proteínas (LBP)
Universidade Federal do Estado do Rio de Janeiro (UNIRIO)
Rio de Janeiro – RJ – Brasil

{fredcs, fporto, anamoura, dpalazzi}@lncc.br,
luis.castro@cob.org.br, abassini@me.com, cameron@unirio.br

Abstract. *Human Centric computing is a new area that places the human being at the heart of systems' functionality. In this work we present a system of this kind that supports the follow-up of high performance athletes. The system is structured on the follow-up of variations of athlete´s observable conditions in time. A mobile object trajectory conceptual model has been adapted to model the variations of observable conditions on a virtual space, which has been named metaphoric trajectories. This system enables registering and analyzing athlete´s metaphoric trajectories of different observable elements, such as heartbeats, glucose, lactose, etc., and even the athlete´s psychological estate. The metaphoric trajectory model supports interesting time-variation analyses that put into perspective the organic behavior of different athletes of the same discipline with impacts on athlete nutrition and training planning. This system is being used by the Brazilian Olympic Committee (COB) and will support Brazilian athlete´s follow-up during the Olympic Games.*

Resumo. *Sistemas computacionais centrados no Ser Humano é uma nova área que coloca o ser humano como elemento central da funcionalidade do sistema. Neste trabalho, apresentamos um sistema que tem por objetivo acompanhar Atletas de alto rendimento. Este sistema tem por objetivo monitorar variações das condições dos atletas relacionadas à algumas de suas características fisiológicas, que variam no tempo. Um modelo conceitual de trajetórias para objetos móveis foi adaptado para as variações das condições observadas em um espaço virtual, aqui denominadas trajetórias metafóricas. Esse sistema permite registrar e analisar as trajetórias metafóricas dos atletas de diferentes elementos, tais como batimento cardíaco, glicose, lactose e mesmo, a variação do estado psicológico do atleta. O modelo de trajetória metafórica suporta análises interessantes que variam no tempo, colocando em evidência o comportamento orgânico de diferentes atletas de mesma modalidade, influenciando o planejamento nutricional e de treinamento dos mesmos. Atualmente, o sistema SAHA está em utilização no Comitê Olímpico Brasileiro.*

1. Introdução

Obter os três primeiros lugares em um pódio olímpico exige mais do que talento e treinamento. Recentemente, o presidente do Comitê Olímpico Brasileiro (COB) declarou seu objetivo de colocar o Brasil, até 2016, entre os dez primeiros no quadro geral de medalhas olímpicas. Para atingir esse objetivo, o COB criou o Laboratório Olímpico (LO) com sede no Rio de Janeiro. O LO é o primeiro Centro Olímpico de Pesquisa Latino-Americano e tem como finalidade integrar diferentes áreas da ciência, visando aplicar técnicas multidisciplinares para aumentar o desempenho dos atletas.

O LO é um exemplo típico do que se tem chamado de Computação Centrada no Homem (Kling 1997). A partir de dados coletados de atletas durante seu período de treinamento, pesquisadores do LO investigam características físico-bioquímicas que possam melhorar seus resultados. Neste sentido, diversas medições são realizadas com o atleta, incluindo a coleta de sangue para análise, anotações da frequência cardíaca e tabelamento do próprio atleta sobre sua alimentação. A visão centrada no atleta pretende integrar e analisar essas informações conforme sua variação durante os treinos e ao longo do período de preparação para competição.

De forma a apoiar a etapa de preparação de um atleta, um *Data Warehouse* (DW) foi projetado com a finalidade de monitorar elementos de interesse observacionais, aqui denominados *elementos observáveis* (EO), através de suas variações no tempo e conforme o estado do atleta. Assim, para que o DW forneça uma visão integrada composta de múltiplos EO, adotou-se um modelo de representação canônico baseado no modelo de trajetórias de objetos móveis Spaccapietra *et al.* (2008). Nesta adaptação, objetos móveis são substituídos pelos EO, cujos valores coletados variam no tempo e em um espaço virtual formado pelo domínio de seus valores, compondo o que chamamos de trajetória metafórica Porto *et al.* (2011). O DW concebido tem como fatos: as medições dos EO, que ao longo de um ciclo de treinamento formam uma trajetória e, dimensões de análise das condições do atleta, a exemplo do nível de glicose no sangue, peso, estado do treinamento, objetivo da avaliação, etc.

A modelagem do DW adotada permite que o sistema realize análises com agregações obtidas diretamente das medições, bem como das agregações de medições em trajetórias. Nestas últimas, o perfil das curvas fornece informações relevantes tanto sobre o comportamento individual do atleta como no aspecto coletivo, quando curvas de um mesmo EO de vários atletas podem ser comparadas.

Neste contexto, este trabalho apresenta o *SAHA*, um sistema que implementa um DW para o acompanhamento holístico de atletas. O esquema do banco de dados foi implementado em PostgreSQL, cujo módulo cliente da aplicação, desenvolvido em Java, permite o cadastramento das dimensões envolvidas e a entrada das medições. Estas últimas são extraídas automaticamente a partir dos resultados de exames dos atletas, ou inseridas manualmente pelo atleta ou por um técnico.

O restante deste trabalho encontra-se organizado em 5 seções, da seguinte forma. A seção 2 contempla os trabalhos relacionados; a seção 3 aborda a formalização do modelo de trajetórias e o modelo do DW; a seção 4 apresenta algumas consultas elaboradas com o sistema, cuja análise dos gráficos gerados levou a resultados importantes sobre os atletas monitorados. Por último a seção 5 tece algumas conclusões sobre o trabalho.

2. Trabalhos Relacionados

Nos últimos anos, vários sistemas foram desenvolvidos com a finalidade de monitorar os atletas. No entanto, os trabalhos encontrados na literatura apontam para trabalhos focados em pontos de observação específicos de atletas. Busso *et. al.* (1997) desenvolveram um modelo de adaptações para o treinamento físico, com base em um algoritmo recursivo de mínimos quadrados. O desempenho (*output*) foi matematicamente relacionado com as cargas de treinamento (*input*), através de uma função de transferência, incluindo dois filtros de primeira ordem, um com um ganho positivo, atribuído para a adaptação ao exercício, e um com o ganho negativo, atribuído ao efeito fatigante das cargas de treinamento. O desempenho do modelo é obtido por convolução das doses de treino, quantificado a partir do nível do exercício e tempo, para a resposta ao impulso.

Já Cormack *et. al.* (2008) propuseram avaliar as variações do estado neuromuscular e hormonal e sua relação com o desempenho ao longo de uma temporada de jogadores de elite do *Australian Rules Football* (ARF). No método desenvolvido, os jogadores realizavam saltos únicos e saltos contra o movimento, e a saliva era coletada para medir os níveis de cortisol e testosterona, antes e durantes os 22 jogos da temporada. A partir dessas informações foram criadas correlações para analisar as relações entre as variáveis da avaliação e a duração do jogo, a carga de treino e o desempenho do atleta.

Até onde pudemos investigar, não foi encontrado na literatura nenhum sistema de acompanhamento de atletas focado na análise de uma variada gama de elementos, que vão desde elementos bioquímicos até condições fisiológicas e nutricionais. O sistema *SAHA*, no entanto, além de ser um sistema Web, se diferencia pelo fato de possibilitar o monitoramento de atletas de alto desempenho, sob diversos aspectos variáveis no tempo.

3. Modelo de Dados Olímpico

O conceito de trajetória tem sido utilizado em um grande número de aplicações que envolvem objetos em movimento, tais como: veículos e pessoas equipados com dispositivos GPS (*Global Positioning System*), encomendas etiquetadas com RFID (*Radio-Frequency Identification*), etc. Nesta seção, nos concentramos em um tipo particular, chamado trajetórias metafóricas, com enfoque nos dados oriundos do acompanhamento de atletas coletados em diferentes ciclos de treinamento.

3.1 Modelo de Trajetórias

O modelo de trajetórias adotado neste trabalho estende o proposto inicialmente por Spaccapietra *et al.* (2008), cujas trajetórias eram analisadas sob a perspectiva de um objeto variando no espaço-tempo. No modelo de trajetória estendido, uma trajetória é definida como *um registro da evolução da posição – percebida como um ponto – de um objeto que se move no espaço durante um dado intervalo de tempo, a fim de atingir um objetivo.* Também pode ser considerado como uma sequência de movimentos que vai de um ponto de parada ao ponto subsequente. Além disso, o conceito de trajetória metafórica surge para definir a evolução de objetos que não estão relacionados ao movimento físico, tal como a trajetória de uma pessoa, ou o processo de crescimento de uma criança, etc.

Assim sendo, nosso modelo de trajetória estendido é composto pelas entidades: *trajetória*, *movimento* e *medição* (ver Figura 1). Nesse contexto, as medidas relacionadas a uma variável particular do atleta é modelada como uma trajetória metafórica em extensão ao senso comum sobre as trajetórias que consideram a variação de um objeto no espaço-

tempo. As trajetórias metafóricas correspondem às variações de uma variável do atleta ao longo de estados em um ciclo de treinamento.

Vale ainda observar que a entidade *trajetória* apresenta uma visão global da mesma, enquanto a *medição* e *movimento* representam a forma pela qual os mesmos são percebidos, i.e., os valores obtidos durante a trajetória. Nesse caso, cada trajetória tem exatamente um início e um fim, um ou mais movimentos e zero ou mais paradas, onde o número de paradas determina o número de movimentos.

3.2 Modelagem do DW

Esta seção descreve a modelagem de dados utilizada na implementação do sistema de acompanhamento de atletas. A Figura 1 representa uma síntese do DW que utiliza as medições e as trajetórias metafóricas como fatos, e como dimensões o estado do treinamento (rotina), o atleta, elemento observável, e as avaliações.

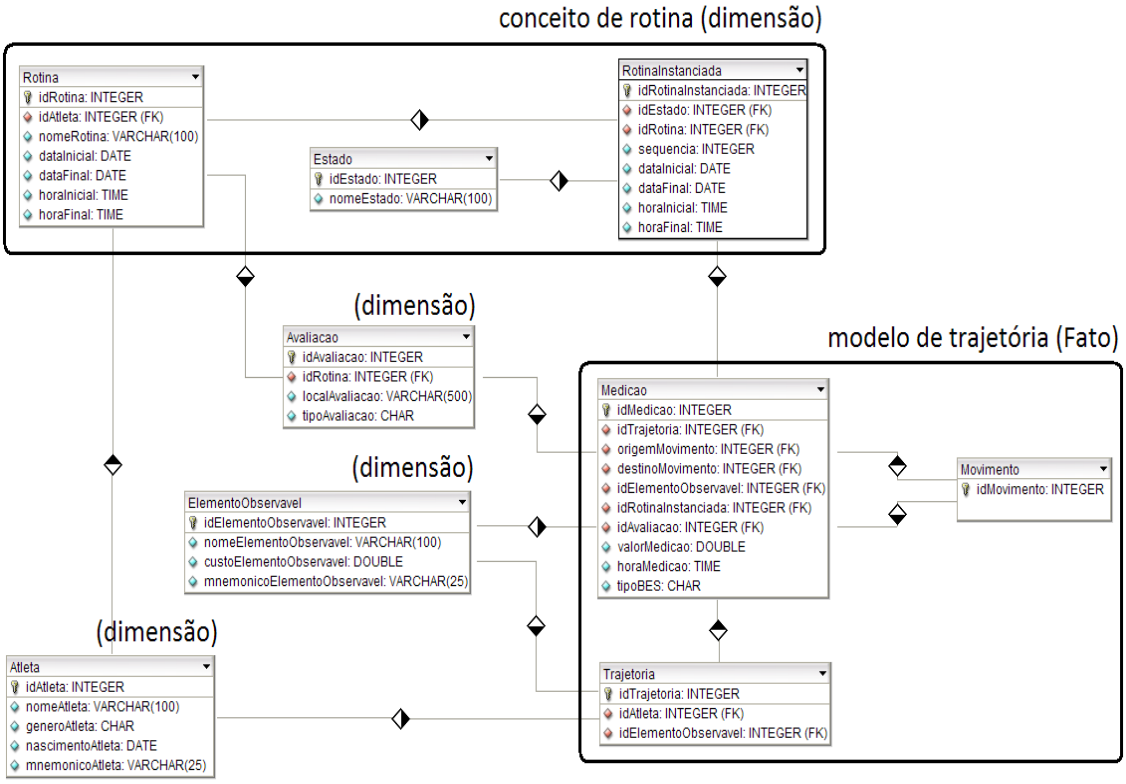


Figura 1. Síntese do modelo de dados

O conceito de rotina, composto pelas entidades: *rotina*, *rotina instanciada* e *estado* (ver Figura 1) é responsável por mapear os ciclos de atividades de um atleta, com informações estratégicas (estados), auxiliando o especialista na avaliação sobre a intensidade de um dado ciclo de treinamento ao qual é submetido. Esses ciclos de treinamento (rotina) contemplam diferentes sequências de acontecimentos (rotinas instanciadas). Além disso, são ordenadas cronologicamente e independentes de medições.

A entidade *elemento observável* (EO) descreve o elemento a ser observado em um atleta, dentre os quais podemos citar: nível de glicose, peso, pressão cardíaca, lesão muscular, tempo gasto para percorrer uma dada distância, etc. A coleta da informação relacionada ao EO, aqui chamada de medição, pode consistir de um único valor (discreto) ou uma sequência de valores que compõem uma trajetória.

O resultado de um EO discreto refere-se a uma ocorrência única dentro de um ciclo pré-definido, como por exemplo: a ingestão de um suplemento nutricional, a ocorrência de uma lesão, etc. Já quando o EO resulta em uma trajetória, é porque a medição do EO ocorreu mais de uma vez dentro de um ciclo pré-definido de tempo. Vale ressaltar que ocorrências discretas de EO iguais poderão ser observadas como trajetórias se os diferentes ciclos em que ocorreram forem "aproximados".

A entidade *atleta* representa o atleta propriamente dito, sendo o provedor de informações analisado sob inúmeras condições e perspectivas. A entidade *avaliação* é responsável por identificar a origem motivadora da medição no qual o atleta está envolvido. Essas informações podem variar de aspectos geográficos (ex.: atleta em competição internacional de Judô – Tóquio – Japão) para detalhes da medição do EO (ex.: EO obtido através de coleta sanguínea).

4. Análise Comparativa de Trajetórias

O modelo de dados tem como principal objetivo abordar o conceito de trajetórias sobre as informações dos atletas. De modo a analisar o modelo quanto a sua capacidade em responder questões relacionadas ao desempenho dos atletas, algumas consultas chaves foram elaboradas. Estas são descritas a seguir, juntamente com uma análise sobre os resultados obtidos.

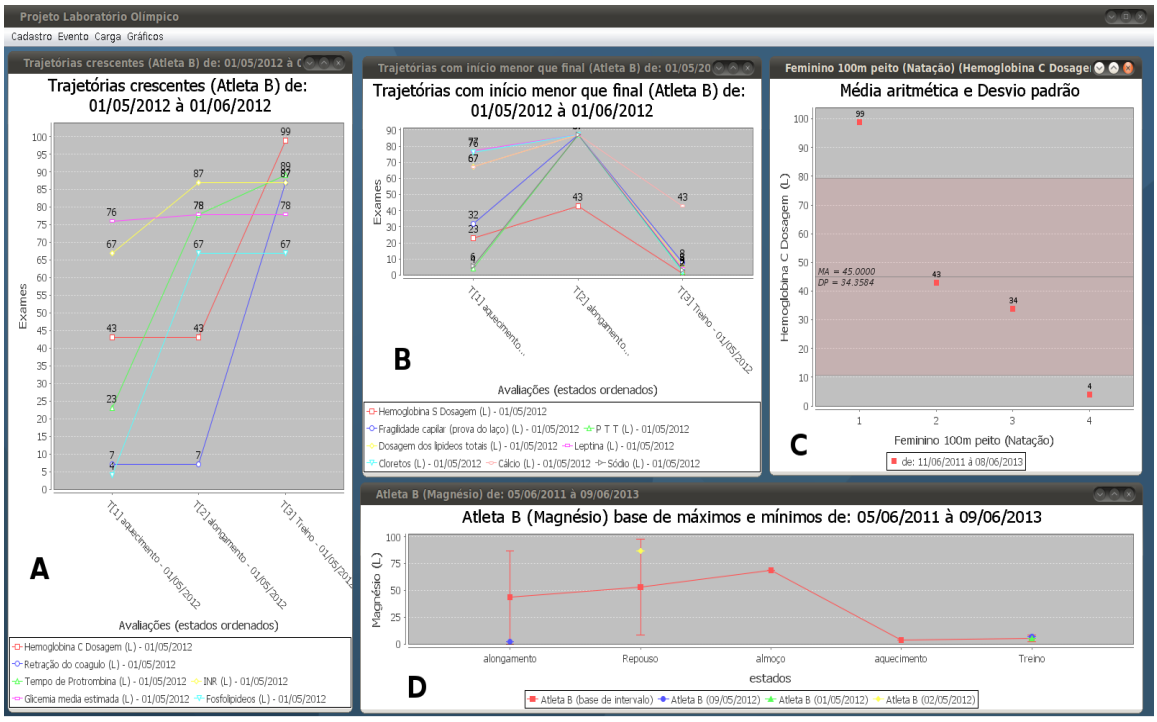


Figura 2. Gráficos resultantes das consultas. (A) Trajetórias crescentes; (B) Trajetórias com extremidade inicial maior que extremidade final; (C) Média aritmética e desvio padrão e (D) Trajetórias encontradas a partir de máximos e mínimos de um atleta base.

Os resultados obtidos na execução das consultas são exibidos na Figura 2. As trajetórias crescentes[1] de um atleta (Figura 2 (A)), consideram todos os exames avaliados em um intervalo de tempo para todos os estados de observação ordenados. De acordo com

[1] O sistema também permite variações decrescentes.

os especialistas, identificar trajetórias cujos valores observados formam uma curva crescente pode apontar para um comportamento anormal de um atleta com respeito a um dado elemento. Outro desdobramento são as trajetórias que apresentam a medição do estado inicial maior que o estado final ou vice-versa (ver Figura 2 (B)), nesse caso, apenas as extremidades da trajetória são comparadas com base nas avaliações para os elementos de observação num período determinado. Esse tipo de avaliação pode contribuir, por exemplo, para indicar se um determinado elemento observável está de acordo com os valores esperados, desprezando os estados intermediários. Os gráficos da média aritmética e desvio padrão (Figura 2 (C)) servem para ilustrar a dinâmica do sistema. Nesse caso, a referida consulta não utiliza o modelo de trajetórias para obter as informações, mas sim, o resultado das medições agrupadas por modalidade esportiva. Essa consulta, aparentemente simples, possibilita a análise do comportamento dos elementos observacionais frente aos diferentes tipos de modalidade permitindo, por exemplo, classificá-las por similaridade quanto ao esforço realizado. Finalmente, com uma maior complexidade, a Figura 2 (D) ilustra os resultados da consulta para aqueles atletas que possuíam ocorrências de valores em estados iguais ao de um atleta base. Os resultados dessas medições estavam entre os valores de máximos e mínimos delimitados pelo atleta base, para um dado EO. Esse tipo de informação resultante é interessantíssimo pois poderá, por exemplo, auxiliar os especialistas na percepção de atletas jovens que possuem perfil aproximado ao de atletas consagrados.

5. Conclusão

Esse trabalho apresentou o sistema SAHA, desenvolvido para monitorar atletas de alto desempenho durante suas fases de treinamento. Baseado na noção de trajetórias metafóricas, esse sistema permite registrar e analisar trajetórias relativas ao treinamento de atletas sob diferentes elementos, permitindo aos especialistas tirarem conclusões importantes sobre os resultados gráficos obtidos. Trata-se de um sistema desenvolvido para a Web e, segundo nosso conhecimento, este é o primeiro sistema a explorar o conceito de trajetórias metafóricas, representando uma contribuição significativa para a área de Computação Centrada no Homem.

Referências

Busso, T., Denis, C. and Bonnefoy, R. (1997). “Modeling of adaptations to physical training by using a recursive least squares algorithm.” Journal of Applied Physiology 82:1685-1693.

Cormack, S. J., Newton, R. U., McGuigan, M. R. and Cormie, P. (2008). “Neuromuscular and Endocrine Responses of Elite Players During an Australian Rules Football Season.” International Journal of Sports Physiology and Performance, 2008, 3, 439-453.

Fabio Porto, Ana Maria de C. Moura, Frederico C. da Silva, Adriana Bassini, Daniele C. Palazzi, Maira Poltosi, Luis Eduardo Viveiros de Castro, L. C. Cameron (2011), A metaphoric trajectory data warehouse for Olympic athlete follow-up, Concurrency and Computation: Practice and Experience, DOI: 10.1002/cpe.1869.

Kling, Rob and Star, Susan Leigh (1997). "Human centered systems in the perspective of organizational and social informatics". Human Center Systems. National Science Foundation.

Spaccapietra S, Parent C, Damiani M, Macedo J, Porto F, Vangenot C. (2008) A conceptual view on trajectories. Data Knowledge Engineering; 65(1):126–146.

Higiia: A Perceptual Medical CBIR System Applied to Mammography Classification*

Marcos V. N. Bedo[1] **, Marcelo Ponciano-Silva**[1,2] **, Daniel S. Kaster**[3] **,**
Pedro H. Bugatti[1,4] **, Agma J. M. Traina**[1] **, Caetano Traina Jr.**[1]

[1]Dept. of Computer Science – University of São Paulo (USP)
P.O.Box 668 – CEP 13.560-970 – São Carlos, SP, Brazil

[2]Fed. Institute of Education, Science and Technology of the Triângulo Mineiro (IFTM)
Av. Barão do Rio Branco, 770 – CEP 38.064-790 – Uberaba, MG, Brazil

[3]Dept. of Computer Science – University of Londrina (UEL)
P.O.Box 6001 – CEP 86051-990 – Londrina, PR, Brazil

[4]Dept. of Computer Engineering – Fed. Technological University of Paraná (UTFPR)
Av. Alberto Carazzai, 1640 – CEP 86300-000 – Cornélio Procópio, PR, Brazil

{bedo, ponciano, pbugatti, agma, caetano}@icmc.usp.br, dskaster@uel.br

Abstract. *Content-Based Medical Image Retrieval (CBMIR) has been used in the medical field to assist several tasks, from training to computer-aided diagnosis. However, there are several gaps that must be filled to approximate the system's retrieval quality and performance over large image databases to the user's similarity perception and needs. This paper presents the Higiia, a modularized CBMIR system that integrates several consolidated techniques that compose the similarity query process, including feature extractors, distance functions and indexing structures for complex data, and allows the user to set custom parameters that best represent his/her perception. Results of experiments at the Clinical Hospital of the USP-RP are also presented, where the system is under test.*

1. Introduction

With the increasing availability of radiological scanners, imaging dianosis has growing at a fast pace. Content-Based Medical Image Retrieval (CBMIR) systems are applications developed to improve the retrieval effectiveness of images generated by radiologic exams based on the images' visual content. CBMIR systems are specially useful in Case-based Medicine, which highly rely on past similar cases for diagnosing a new one. Similarity queries are typical in these systems, such as: "which are the patient studies that contain the RX images most similar to the RX image of a given (undiagnosed) patient"?

This scenary highlights two main challenges: (i) to improve the existing techniques to help specialists to search and analyze medical images based on their content, and (ii) to provide methods to support such analysis over the huge medical image databases maintained by the healthcare institutions. The first challenge is referred in the literature as the Semantic Gap, which is the mismatch between the user notion of similarity and the similarity computed by the CBMIR systems relying on features automatically extracted from the images [Güld et al. 2007]. One effort to mitigate the semantic gap is to introduce

*This research was supported by FAPESP, CNPq and CAPES.

aspects in the CBMIR system that reflect the user perception, which is highly dependent on the application's context. The second challenge demands employing a robust mechanism to store/manage images and specialized indexing structures and algorithms to query large medical image databases.

This paper presents the Higiia, which is a CBMIR software including support for perceptual retrieval of medical images. The main strengths of Higiia are that it natively supports images in the DICOM format (the standard in the medical field), it allows handling subsets of the enterprise medical image database, building distinct medical contexts through filters over the DICOM metadata, it allows setting up perceptual parameters on-the-fly, adjusting the similarity evaluation to the specific image context, and it relies on a fast and robust retrieval engine executed over a DBMS. The paper also presents an interface implemented on Higiia for assisting specialists in the classification of mammograms. This Higiia instantiation is being used in experiments at the Clinical Hospital of the Faculty of Medicine of Ribeirão Preto of the University of São Paulo.

2. Related Work

There are several works employing content-based image retrieval tecniques in the medical field [Deserno et al. 2009, Akgül et al. 2011]. Despite of the advances in CBMIR there still are many challenges for including this technology in the clinical pratice. From the medical point of view, several studies, such as [Depeursinge et al. 2011], suggest that one of the main limitations the CBMIR paradigm is the lack of interactive tools that allow the specialist to test and validate what is really similar regarding his/her perception. This kind of application requires putting together knowledge of computer science and of medicine/radiology and it is still demanded to bridge several gaps between the user expectation and the system results.

Deserno et al. suggested in [Deserno et al. 2009] a systematic view of gaps in CBMIR research, which organize them in four broad categories: content gaps, features gaps, performance gaps and usability gaps. The content gaps encompass the user's understanding of image and the clinical context of use of the system. The features gaps include the methods employed to extract and represent numerically the image features and their limitations. The performance gaps address the answer speed and the integration with other patient care information systems. Finally, the usability gaps include the query types available and the system ability of learning from user feedback. Analyzing the existing works in CBMIR according to this systematic view of gaps, it can be noticed that, in general, they focus either on the semantics/quality of the results (usually providing at most a rough description of the implementation details) or in the system/query performance. Examples include the SMIRE (Similar Medical Image Retrieval Engine) [Cheng et al. 2005] and the FIRE (Flexible Image Retrieval Engine) system [Deselaers et al. 2004]. On the other hand, there are other systems, such as MedFMI-SiR [Kaster et al. 2011], which present high flexibility and performance, being adequate to handle huge image databases. However, these systems only provide the basis of CBMIR, requiring the development of specialized features and of the end user applications.

The CBMIR system presented in this paper covers gaps in breadth. Regarding the content and the features gaps, our system is able to define semantic contexts through the definition of perceptual parameters. Humans consider several visual patterns when com-

paring images, associating different relevances to these patterns according to their perception and background. *Perceptual parameters* are keywords in the application domain that aim at discriminating which visual characteristics have stronger/subtler influence in the user's notion of similarity, allowing defining the combination between the feature extractor and the distance function, as well as their fine tune settings, that best represents the user perception [Ponciano-Silva et al. 2009]. Regarding the performance gaps, our system adheres to standards, handling DICOM images natively and relying on a robust and efficient SQL-based storage system, as its backend is based on the MedFMI-SiR module over the Oracle DBMS. Finally, regarding to the usability gaps, our system is able to answer queries using DICOM metadata and/or similarity conditions and it provides relevance feedback and query refinement methods. These characteristics allow joining several techniques to provide retrieval interfaces for several medical routines, as the interface for mammogram classification enhanced with CBMIR presented herein.

3. The Architecture of the Higiia System

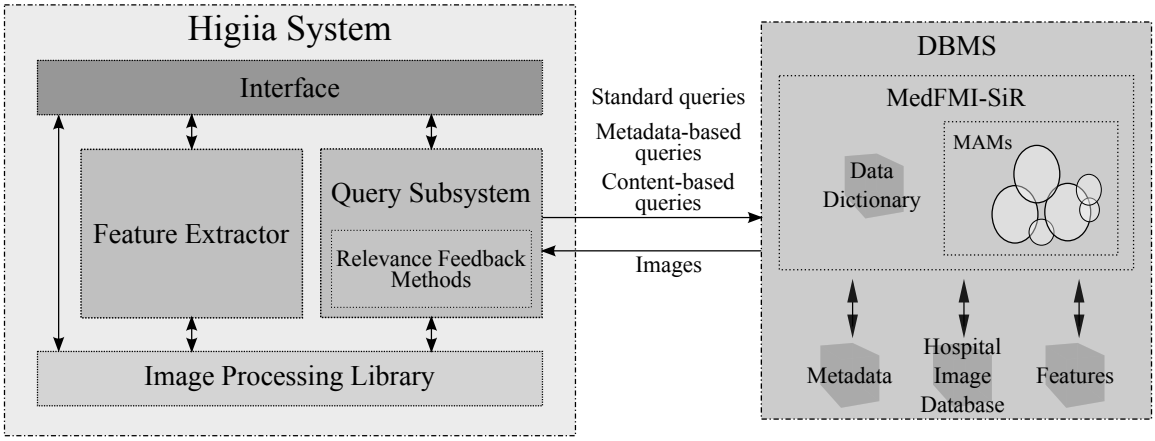


Figure 1. Overview of the Higiia system's architecture.

The Higiia system is composed by a set of modules, as illustrated in Figure 1. The Feature Extractor is the module responsible for extracting the visual features from the images. The Feature Extractor interacts with the Image Processing Library, which provides a common interface for image processing and presentation that allow attaching third party image processing libraries. Currently the Image Processing Library interacts with the libraries DCMTK[1] for manipulating DICOM images and OpenCV[2] for other formats. The Query Subsystem receives the queries from the user interface, construct the statements and invoke the DBMS to execute them. The hospital image database and the image features are stored in the DBMS as BLOB columns, and the associated metadata are stored using traditional or XML data types. The query types supported include standard queries (i.e. queries employing conventional attributes), DICOM metadata-based queries and content-based queries. The content-based queries are executed over Metric Access Methods (MAMs) using the MedFMI-SiR module installed on the DBMS, providing an efficient and scalable retrieval. The Query Subsystem include Relevance Feedback

[1]http://dicom.offis.de/dcmtk.php.en
[2]http://opencv.org

methods for interactively tuning the queries to achieve better results. The Query Subsystem also collects information during the users' interactions, including the perceptual and the other query parameters, the returned images indicated as relevant by the user in the feedback cycles and so on. Such information compose the user profiles, which can be used for improving the similarity computation according to each user perception. A few feature extractors and several distance functions are available, allowing setting up various perceptual parameters for different image contexts. Finally, the Interface implements the end user interaction, communicating with the other modules. The tool was coded in C++ using the cross-platform Qt framework[3] to provide portability.

4. The Higiia's Mammogram Classification Interface

This section describes an interface developed to evaluate the effectiveness of using CBMIR for supporting mammogram classification. The problem statement is as follows. There is an image from a mammography exam to be classified (diagnosed) according to the type of lesion (e.g. calcification or mass), the BI-RADS category, the breast parenchyma density and the subtlety of the lesion. It is desired that the specialist (a radiologist or a radiology resident) is allowed to search for past diagnosed images that are visually similar to the image to be classified for supporting the current diagnosis. The hypothesis is that providing similar classified examples the rate of correct diagnosis and/or the specialist sureness can be improved.

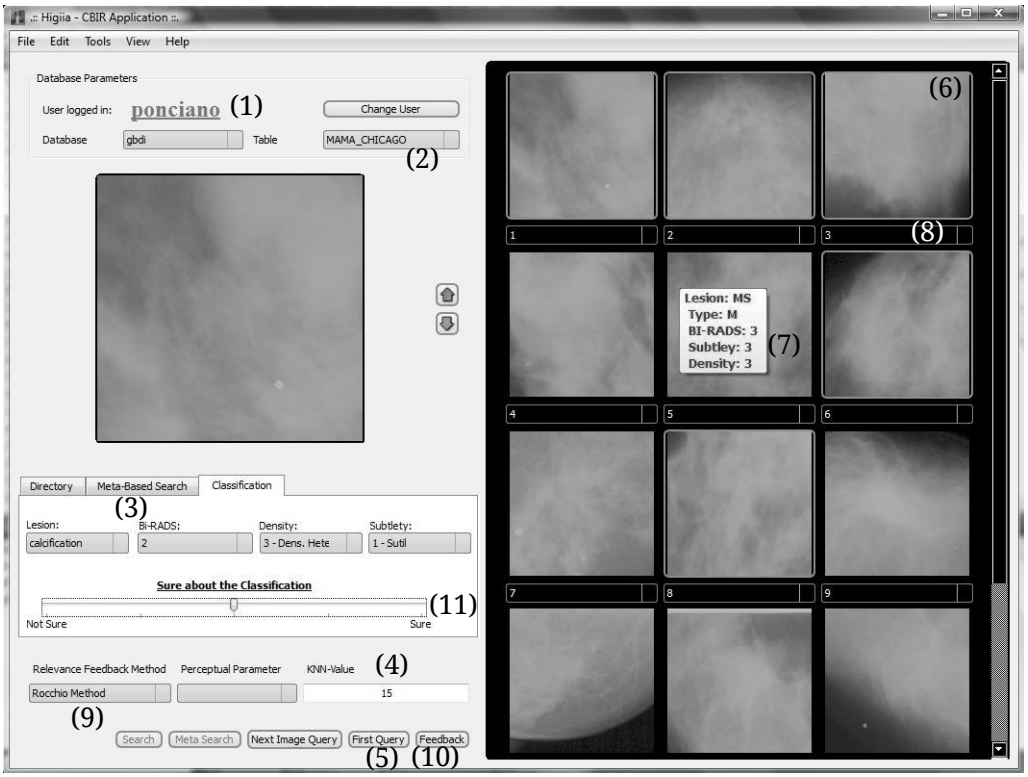


Figure 2. The Mammogram Classification Interface of Higiia.

We developed the Mammogram Classification Interface on Higiia to address the stated problem. Figure 2 shows a screenshot of the interface. The user must log in (step 1

[3]http://qt.nokia.com

in the figure) and select the desired database and table (step 2), specifying the search context. Once logged in, every user action is recorded in his/her profile. The profile information is devoted to further user intent characterization and statistical validation of the tool. Afterwards, the user loads the undiagnosed image and classify it (step 3). Usually, the undiagnosed image is already stored in the database, as the screening equipments send the generated images to the hospital's Picture Archiving and Communication System (PACS), which manages the image repository. The PACS can rely on the DBMS to store the images directly on the database, or else they can be imported from the PACS server filesystem into the database by a deamon running in the PACS server. Therefore, the image to be classified is usually obtained from a filesystem directory or through DICOM metadata-based search (these search tabs are hidden in the figure). In the figure, the image to be classified is shown in the left of the screen and the classification controls are below it. At this point, the user may want to search for similar past cases, using the undiagnosed image as the query reference. To do so, he/she must define the query parameters (step 4), that are the number of images to be retrieved (the k value, as the system will perform a k-Nearest Neighbor query) and the desired perceptual parameter, and submit the first query (step 5). If the user omits any of these parameters, the system will use the default value previously defined for the search context. The Query Subsystem prepares the query and submits it to the DBMS using the extended SQL provided by MedFMI-SiR. Considering that the query image is on the database, the query is stated using a subquery to get the image query features, as follows:

```
SELECT image FROM image_table WHERE euclidean_knn(image_sign,
  (SELECT image_sign FROM image_table WHERE id = query_image_id)) <= k;
```

where `image` is the attribute of the `image_table` that stores the images; `euclidean_knn` is the extended operator that runs a k-NN query using the `Euclidean` distance, whose parameters are the attribute of the input table that stores the extracted image features (the image signature `image_sign`), the query reference, which is the image signature returned by the subquery, and the number of images (neighbors) to be returned (`k`). Both the metadata-based queries and the content-based queries are executed over indexes. The indexes on the DICOM metadata tags and the MAM indexes on the signature attributes are created by Higiia for each query context of interest. If the query image is not on the database, the system extract its visual features and saves the generated signature in a temporary BLOB variable, which is provided as the query reference, in place of the subquery in the presented example. The query result is shown through thumbnails of the retrieved images in the right of the screen, in decreasing order of similarity (step 6). The user can visualize the diagnosis of the retrieved images by positioning the mouse over them (step 7). Users can also hide images judged as irrelevant right clicking on them, as during interviews some radiologists indicated that this action would help to reduce the visual noise produced by such images in the result set. If the user wants to refine the search, it can be done through a relevance feedback iteration. First, the user informs which are the relevant retrieved images, providing the perceived degree of relevance for each them in the combo box just below the image (step 8). Afterwards, the user chooses the relevance feedback method, such as the Rocchio algorithm (step 9), and presses the Feedback button (step 10), triggering the process. Users can use as many relevance feedback cycles as needed to improve the acuracy and the certainty of his/her diagnosis, which must also be indicated by the user to be recorded as part of the classification (step 11).

The Higiia's Mammogram Classification Interface is under testi in a real medical environment at the Clinical Hospital of the Faculty of Medicine of Ribeirão Preto of the University of São Paulo (HCFMRP-USP). An initial experiment was conducted with 10 specialists (radiologists and radiology residents/technicians) that were asked to classify selected mammograms, first just analysing the image and afterwards using our interface to check the classification of previously diagnosed similar images. Users employed a perceptual parameter that maps to a 256-bin grayscale normalized histogram image signature and to the Euclidean distance function. The sucess rate without using the system was high, being up to 88% in the average. However, it was verified that the interface helped the specialists, raising the classification success rate to more than 93%. Although the improvement is not so impressive, it has clinical relevance as it corresponds to difficult cases. Moreover, the classification confidence also increased with the aid of CBMIR.

5. Conclusions

This paper presented the Higiia system, which assembles several techniques developed during the progress of the research in similarity retrieval. Higiia is composed by modules that are integrated to allow filling different categories of gaps identified in CBMIR systems. It employs feature extractors, distance functions, metric access methods, relevance feedback techniques and image processing functions, supporting several image formats, including DICOM. The paper also presented the Mammogram Classification Interface, which aims at assessing specialists in diagnosing mammograms and is being tested by radiologists in a clinical hospital, having presented interesting initial results.

References

Akgül, C., Rubin, D., Napel, S., Beaulieu, C., Greenspan, H., and Acar, B. (2011). Content-based image retrieval in radiology: Current status and future directions. *Journal of Digital Imaging*, 24:208–222.

Cheng, P.-C., Chien, B.-C., Ke, H.-R., and Yang, W.-P. (2005). SMIRE: Similar medical image retrieval engine. In *Inf. Access for Text, Speech and Images*, pages 918–918.

Depeursinge, A., Fischer, B., Müller, H., and Deserno, T. (2011). Prototypes for content-based image retrieval in clinical practice. *The Medical Open Info Journal*, 5(1):58–72.

Deselaers, T., Keysers, D., and Ney, H. (2004). FIRE - Flexible Image Retrieval Engine: ImageCLEF 2004 evaluation. In *CLEF*, pages 688–698.

Deserno, T. M., Antani, S., and Long, R. (2009). Ontology of gaps in content-based image retrieval. *Journal of Digital Imaging*, 22(2):202–215.

Güld, M. O., Thies, C., Fischer, B., and Lehmann, T. M. (2007). A generic concept for the implementation of medical image retrieval systems. In *Int. Journal of Medical Informatics*, volume 76, pages 252–259. Elsevier.

Kaster, D. S., Bugatti, P. H., Ponciano-Silva, M., Traina, A. J. M., Azevedo-Marques, P. M., Santos, A. C., and Jr., C. T. (2011). MedFMI-SiR: A powerful DBMS solution for large-scale medical image retrieval. In *ITBAM*, pages 16–30.

Ponciano-Silva, M., Traina, A. J. M., Azevedo-Marques, P. M., Felipe, J. C., and Jr., C. T. (2009). Including the perceptual parameter to tune the retrieval ability of pulmonary CBIR systems. In *CBMS*, pages 1–8.

WED-tool: uma ferramenta para o controle de execução de processos de negócio transacionais*

Marcela O. Garcia[1], **Pedro Paulo de S. B. da Silva**[1],
Kelly R. Braghetto[2], **João E. Ferreira**[1]

[1]Instituto de Matemática e Estatística – Universidade de São Paulo
(IME-USP) – São Paulo – SP – Brasil

[2]Centro de Matemática, Computação e Cognição – Universidade Federal do ABC
(CMCC-UFABC) – Santo André – SP – Brasil

{mortega,pedro,kellyrb,jef}@ime.usp.br

Abstract. *WED-flow (Workflow, Event, Data-flow) is a data-oriented approach, created to allow management and incremental modeling of business processes. Combining several Advanced Transaction Models, the WED-flow approach guarantees consistent execution of instances and also provides a differentiated exception handling. In this paper we present the main aspects of* ***WED-tool****, a software tool which implements this approach in a concise and intuitive manner.*

Resumo. *WED-flow (Workflow, Event, Data-flow) é uma abordagem orientada a dados, criada para permitir o gerenciamento e a modelagem incremental e evolutiva de processos de negócios. A combinação de diversos Modelos Transacionais Avançados garante a execução consistente das instâncias, assim como fundamenta seu tratamento diferenciado de exceções. Neste trabalho, apresentamos os principais aspectos da ferramenta de software* ***WED-tool****, que implementa essa abordagem de maneira concisa e intuitiva.*

1. Introdução

Com o crescimento e popularização da Internet, atividades que antes eram mais comumente realizadas manualmente receberam formas virtuais de execução graças aos sistemas web. Com a necessidade de se projetá-los, a utilização de **processos de negócio** e consequentemente a utilização de sistemas de **gestão de processos de negócio** (GPN) cresceu rapidamente e trouxe à tona uma série de novos desafios. Como tais, podemos citar a dificuldade de se evoluir o modelo de um processo de negócio, a garantia de propriedades transacionais à execução de instâncias, o tratamento de exceções e a execução concorrente de instâncias. Várias abordagens foram propostas visando enfrentar esses desafios [Garcia-Molina and Salem 1987, Schuldt et al. 2002, Vidyasankar and Vossen 2011, Bhiri, S. et al. 2011], entretanto, apesar de suas contribuições, ainda há muito a ser feito.

Neste trabalho, apresentamos a **WED-tool**, uma ferramenta para GPN que permite a modelagem de processos de negócio de forma declarativa, a instanciação desses processos e o controle de execução dessas instâncias. A ferramenta flexibiliza o processo de modelagem e de tratamento de exceções sem, no entanto, abrir mão da garantia de propriedades transacionais à execução. A WED-tool é uma implementação da abordagem **WED-flow** apresentada em [Ferreira et al. 2010] e estendida em [Ferreira et al. 2012].

*Este trabalho recebe o apoio financeiro do CNPq no programa bolsa instituicional e da FAPESP por meio do projeto de pesquisa nº 2010/15493-4 e 2011/24114-0.

2. Abordagem WED-flow

Um *processo de negócio* é um conjunto de uma ou mais tarefas estruturadas de forma a realizar um objetivo de negócio específico. O modelo de um processo de negócio, que é usado como base para a criação e gerenciamento de instâncias de processos de negócio, define os passos de negócio (tarefas), as precondições para a execução dos passos e as condições de início e término de cada instância. Cada execução de um processo de negócio gera uma instância e cada instância executa os passos de negócio definidos no modelo, alterando, assim, seu estado até que a condição de término seja alcançada.

A abordagem **WED-flow** (Work, Event processing, Data-flow) foi proposta como uma forma de se modelar processos de negócios, de se instanciá-los, de se executar suas instâncias garantindo propriedades transacionais e de se tratar exceções de uma maneira mais simples. Como nessa abordagem a modelagem dos processos se baseia na perspectiva de orientação a dados, os passos de negócio, chamados de **WED-transitions**, e suas precondições de execução, chamadas **WED-conditions**, operam sempre sobre os estados de dados das instâncias. Tais estados, chamados **WED-states**, são valores para um conjunto de atributos de interesse da aplicação, chamados **WED-attributes**. Dessa forma, a execução de uma WED-transition em uma dada instância de processo só é iniciada se o estado corrente da instância satisfaz a WED-condition associada à WED-transition. Essa última, quando executada, alterará o estado da instância.

Um WED-trigger é um par (WED-condition, WED-transition), que associa uma precondição a um passo de negócio. Um processo de negócio é modelado por meio de um WED-flow, que é um conjunto de WED-triggers mais duas WED-conditions que especificam a condição de início e término da execução de uma instância do processo em questão. Como uma dada aplicação pode ser constituída por diversos processos de negócio, podemos ter diversos WED-flows definidos sobre um mesmo conjunto de WED-attributes.

Alterações nos WED-states geram eventos que são capturados pelas WED-triggers que verificam suas WED-conditions associadas e, caso elas sejam satisfeitas, disparam suas WED-transitions associadas, de forma similar ao modelo ECA (*Event*, *Condition*, *Action*). A execução de uma WED-transition, a qual é modelada como um SAGA Step [Garcia-Molina and Salem 1987], gera um novo WED-state que por sua vez gera um novo evento e assim por diante, até que um WED-state gerado satisfaça a condição de término do WED-flow. Caso alguma inconsistência seja detectada, os mecanismos de recuperação interrompem a execução da instância para tentar retorná-la a um estado consistente e dar prosseguimento à sua execução. A definição de inconsistência será abordada na Seção 3.3.

Durante a modelagem, o projetista define de forma declarativa as WED-transitions, WED-conditions, WED-triggers e WED-flows além das restrições de integridade da aplicação, chamadas de **AWIC** (Application-Wide Integrity Constraints), que são expressas na forma de WED-conditions e impostas sobre os WED-states.

3. Descrição da ferramenta

Nesta seção descreveremos a WED-tool[1], uma implementação da abordagem WED-flow. Essa ferramenta foi implementada usando a linguagem *open-source* Ruby, que é portável e também dá suporte ao uso de diversos tipos de bancos de dados. Considerando suas funcionalidades, a ferramenta pode ser dividida em três partes que serão descritas a seguir.

[1]Mais detalhes da ferramenta estão disponíveis em `http://www.data.ime.usp.br/wedflow`.

3.1. Definição e Manutenção da Estrutura dos Processos de Negócio

As fases da modelagem de um processo de negócio utilizando a abordagem WED-flow foram detalhadamente descritas em trabalhos prévios [Ferreira et al. 2010, Ferreira et al. 2012] e, uma vez que o modelo foi produzido, este deve ser traduzido para uma linguagem concreta de especificação. Nesta implementação da abordagem WED-flow, a especificação do processo de negócio deve ser descrita em um arquivo XML que segue uma estrutura que definimos em um XML Schema. Concretamente, WED-attributes, WED-conditions, WED-transitions, WED-triggers e WED-flows devem ser descritos em XML. Dessa maneira, o arquivo XML produzido será interpretado pela nossa ferramenta, possibilitando a criação da estrutura necessária para controlar a execução do processo de negócio.

Nossa implementação é baseada em um banco de dados relacional e cada elemento do modelo WED-flow está associado a uma relação do banco. De acordo com a definição dos WED-attributes, a relação responsável pelo armazenamento de todos os estados de dados é criada. Assim, cada WED-state é uma instância nessa relação. As definições de WED-conditions, WED-transitions, WED-triggers e WED-flows extraídas do arquivo XML também são armazenadas no banco. Para cada WED-transition definida em um XML fornecido como entrada para a ferramenta, a WED-tool cria um protótipo de duas classes em Ruby: uma para o código que será executado pela ferramenta no disparo da transição, e outra para o código da compensação associada à transição. O projetista do WED-flow é quem deve definir esses códigos, respeitando uma interface bem definida. Com isso, uma transição pode ser qualquer código que possa ser especificado em Ruby, o que inclui chamadas a componentes ou serviços Web, ou até mesmo instruções SQL.

É importante notar que nosso sistema é capaz de controlar diferentes WED-flows para uma mesma aplicação. Enquanto WED-triggers são específicos e diretamente associados a um WED-flow, as definições de WED-conditions e WED-transitions são gerais, facilitando o reuso das mesmas por diversos WED-flows. Além disso, o modelo de um WED-flow pode ser alterado incrementalmente por meio da adição ou remoção de WED-conditions, WED-transitions e WED-triggers, simplificando sua evolução.

3.2. Controle de execução

Após a leitura da especificação do modelo e criação da estrutura inicial, o sistema permite criar instâncias de WED-flows e controlar a execução das mesmas. Nesta implementação, para dar início à execução de uma instância do processo de negócio, um usuário precisa fornecer valores iniciais para criar um WED-state e também selecionar um WED-flow que será instanciado para manipular o novo estado inicial gerado.

Na abordagem WED-flow, o controle de execução dos processos de negócio é determinado por WED-conditions que são verificadas sobre os valores dos atributos dos WED-states. Assim, todos os WED-states devem ser monitorados e quando um deles satisfaz uma condição, a transição associada será disparada assincronamente. Para realizar o monitoramento de estados, a WED-tool cria, para cada WED-trigger definido pelo usuário, uma estrutura que mantém uma fila de WED-states que precisam ser processados e também um intervalo de tempo que indica com que frequência o processamento deve ocorrer. Dessa maneira, cada WED-trigger é responsável por verificar sua própria condição e disparar sua transição se necessário. Sendo assim, após a instanciação do WED-flow, a execução é iniciada por meio do oferecimento do WED-state inicial para

todos os WED-triggers associados ao WED-flow selecionado, ativando a avaliação de WED-conditions que irão habilitar o disparo de WED-transitions.

Cada execução de WED-transition produz um novo WED-state que, por sua vez, pode disparar a execução de outra(s) WED-transitions do WED-flow, dando prosseguimento à execução da instância. O novo WED-state é gerado pela transição por meio da atualização de um conjunto específico de atributos que deve ser definido no código de execução da WED-transition. Esse conjunto também determina para quais WED-triggers o novo estado da instância será oferecido. O novo estado deve ser oferecido para um WED-trigger se, e somente se, a WED-condition associada possui pelo menos um predicado definido sobre um atributo que foi atualizado na produção do novo estado. O oferecimento seletivo foi desenvolvido para evitar o redisparo não-intencional de transições.

Em nossa implementação, além de armazenar todos os estados de dados gerados na instância, mantemos um histórico de todas as execuções de WED-transitions, incluindo detalhes como o estado responsável pelo disparo, estado produzido e data de início e término da execução. Essas informações são fundamentais para o módulo de recuperação agir na ocorrência de exceções e interrupções. Adicionalmente, outras informações referentes ao controle de execução (como por exemplo, as filas dos WED-triggers) são também mantidas em banco de dados para fins de integridade. Dessa maneira, garantimos que, em caso de queda do sistema, as execuções das instâncias poderão ser retomadas.

3.3. Recuperação

Uma instância do WED-flow está inconsistente quando um WED-state **inconsistente** é produzido ou quando há alguma **falha** durante a execução de uma WED-transition. Consideramos inconsistente o WED-state que (i) não respeita as restrições AWIC definidas no modelo, (ii) não satisfaz nenhum WED-trigger e (iii) em sua instância não há mais WED-transitions sendo executadas. WED-states inconsistentes são, normalmente, consequência de falhas de projeto ou de problemas estruturais, como o corrompimento do disco rígido, por exemplo. Uma WED-transition pode falhar devido a problemas de implementação, *time-out* da execução, conflitos de escrita causados por WED-transitions sendo executadas paralelamente, cancelamento explícito da execução e devido a problemas estruturais que resultam na interrupção inesperada da execução, como no caso de uma queda de energia, um defeito em componente do servidor, etc. A detecção da inconsistência de uma instância do WED-flow, independentemente da causa, resulta na ativação do **gerenciador de recuperação** da WED-tool, o qual imediatamente a interrompe.

O objetivo do gerenciador de recuperação é fazer com que a instância interrompida volte a ser consistente. Para isso, por meio do histórico de execuções mantido pela WED-tool, a situação de interrupção é externalizada e a intervenção de um administrador do sistema se faz necessária. É ele, que através de um arquivo XML em formato semelhante ao usado para a definição de processos de negócios, fará a escolha do(s) método(s) de recuperação a ser(em) utilizados. Pode-se dividir os métodos de recuperação em dois tipos: (i) com WED-state consistente e (ii) com WED-state inconsistente. O primeiro tem como característica métodos que sempre partem de um WED-state consistente. Ele compreende **compensações**, chamadas de **WED-compensations**, que desfazem semanticamente uma WED-transition e as **ressubmissões** que ressubmetem um determinado WED-state aos WED-triggers do WED-flow no escopo da instância paralisada. O segundo compreende transições especiais, definidas pelo administrador do sistema, que, a partir de um WED-

state inconsistente, geram um WED-state consistente. É importante salientar que pode haver combinações dos métodos, como, por exemplo, WED-compensations associadas a ressubmissões, etc. Definidos o(s) método(s) de recuperação e seus possíveis parâmetros de execução, o gerenciador de recuperação se encarrega de executá-lo(s). Outra vez consistente, a instância do WED-flow pode dar prosseguimento à sua execução.

A intervenção de um administrador do sistema, prevista no WED-flow é justificada por dois motivos principais: a característica semântica do tratamento de exceção e a necessidade de se diminuir a complexidade da modelagem inicial dos processos de negócio. É muito comum que um mesmo problema tenha diversas soluções e que a qualidade ou eficácia de cada solução varie conforme o contexto em que o problema está inserido. No momento da modelagem inicial de um processo de negócio é geralmente complicado, ou até mesmo impossível, modelar todos os problemas, seus possíveis contextos e suas múltiplas formas de tratamento. Com a intervenção de um administrador do sistema, o melhor método de recuperação é definido e então executado **somente quando ele se faz necessário**. Similarmente, é permitido ao administrador do sistema adicionar, modificar ou remover WED-transitions, WED-conditions, WED-triggers e restrições AWIC também no momento da recuperação. Isso afetará todas as instâncias dos WED-flows relacionados e assim, com essa possibilidade de se adicionar novas regras de negócio e de se modificar as já existentes, diminui-se a complexidade da modelagem inicial e ao mesmo tempo permite-se um tratamento de exceções muito mais preciso.

4. Exemplo

Para ilustrar como nossa implementação funciona, utilizaremos um exemplo de uma agência de viagens, também apresentado em [Ferreira et al. 2012]. Nessa agência, quando um pedido de viagem é feito, o primeiro passo é validá-lo. Após a validação, a reserva de hotel e a compra de passagem aérea podem ser realizadas paralelamente. Se ambas forem executadas com sucesso, o pedido é concluído. A especificação desse processo de negócio, usada como entrada para a WED-tool, é descrita em um arquivo XML [2].

Criamos duas instâncias do processo da agência de turismo. Na primeira (Figura 1[a]), com estado inicial 1.1, inicialmente o pedido é validado (WED-transition$_1$), gerando o estado 1.2, e então as transições de reserva de hotel (WED-transition$_2$) e compra de passagem de avião (WED-transition$_3$) são habilitadas pelo estado 1.2 e disparadas paralelamente. A reserva é realizada (estado 1.3) e a passagem comprada (estado 1.4). O estado 1.4 habilita a execução da WED-transition$_4$, que finaliza o pedido. Para essa instância, as execuções dos passos do processo de negócio são realizadas com sucesso e, consequentemente, o cliente recebe uma reserva de hotel e uma passagem de avião.

Já na execução da segunda instância (Figura 1[b]), inicialmente o pedido é validado e então as transições de reserva de hotel e compra de passagem de avião são disparadas. A primeira é realizada com sucesso, porém ocorre um problema com a segunda e esta é interrompida por *time-out*. Com isso, supondo que o cliente desistiu da viagem, os passos que haviam sido realizados com sucesso precisam ser compensados, retornando a instância para um estado equivalente ao inicial. Na Figura 1[b], a WED-transition$_3$ está representada em cor cinza, indicando que a transição foi disparada mas não concluída. Além disso, os dois últimos estados (2.4 e 2.5) são frutos da execução de WED-compensations, associadas às WED-transitions que já haviam sido executadas na instância.

[2] O XML do exemplo está disponível em `http://www.data.ime.usp.br/wedflow`.

(a)

Estado	Cliente	Pedido	Hotel	Hotel id	Passagem	Passagem id
1.1	Não validado	Recebido	Não requisitado		Não requisitado	
1.2	Validado	Validado	Requisitado		Requisitado	
1.3	Validado	Validado	Reservado	11	Requisitado	
1.4	Validado	Validado	Reservado	11	Comprado	22
1.5	Validado	Finalizado	Reservado	11	Comprado	22

WED-Transition$_1$
WED-Transition$_2$
WED-Transition$_3$
WED-Transition$_4$

(b)

Estado	Cliente	Pedido	Hotel	Hotel id	Passagem	Passagem id
2.1	Não validado	Recebido	Não requisitado		Não requisitado	
2.2	Validado	Validado	Requisitado		Requisitado	
2.3	Validado	Validado	Reservado	33	Requisitado	
2.4	Validado	Validado	Requisitado		Requisitado	
2.5	Não validado	Recebido	Não requisitado		Não requisitado	

WED-Transition$_1$
WED-Transition$_2$
WED-Transition$_3$
WED-Compensation$_2$
WED-Compensation$_1$

Figura 1. (a) Execução da instância 1. (b) Execução da instância 2.

5. Considerações Finais

Embora muitos trabalhos teóricos recentes apresentem propostas de modelos transacionais para processos de negócio [Schuldt et al. 2002], [Vidyasankar and Vossen 2011], [Bhiri, S. et al. 2011], poucos desses modelos foram implementados, o que, em alguns casos, pode indicar uma inviabilidade prática. Este artigo apresenta a ferramenta WED-tool que, utilizando os conceitos da abordagem WED-flow, garante uma modelagem evolutiva, o controle transacional e a diminuição da complexidade do tratamento de exceções em processos de negócio. Na WED-tool, é fácil alterar o modelo de um processo de negócio em qualquer fase do seu ciclo de vida. Os passos de negócio (encapsulados como WED-transitions) são implementados como SAGA Steps, o que garante propriedades transacionais aos processos. O tratamento de exceções é *ad hoc*, permitindo aos projetistas que se concentrem na definição das regras de negócio da aplicação e diminuindo a incidência de erros, uma vez que, no momento da modelagem, nem sempre eles têm clareza de todos os caminhos de execução possíveis.

Referências

[Bhiri, S. et al. 2011] Bhiri, S. et al. (2011). Ensuring customised transactional reliability of composite services. *J. Database Manag.*, 22(2):64–92.

[Ferreira et al. 2012] Ferreira, J. E., Braghetto, K. R., Takai, O. K., Malkowski, S., and Pu, C. (2012). Transactional recovery support for robust exception handling in business process services. In *Proc. of the 19th Int. Conference on Web Services*, pages 303–310.

[Ferreira et al. 2010] Ferreira, J. E., Takai, O. K., Malkowski, S., and Pu, C. (2010). Reducing exception handling complexity in business process modeling and implementation: the wed-flow approach. In *Proc. of the 18th Int. Conference on Cooperative Information Systems*, pages 150–167.

[Garcia-Molina and Salem 1987] Garcia-Molina, H. and Salem, K. (1987). Sagas. *SIGMOD Rec.*, 16:249–259.

[Schuldt et al. 2002] Schuldt, H., Alonso, G., Beeri, C., and Schek, H.-J. (2002). Atomicity and isolation for transactional processes. *ACM Trans. Database Syst.*, 27(1):63–116.

[Vidyasankar and Vossen 2011] Vidyasankar, K. and Vossen, G. (2011). Multi-level modeling of web service compositions with transactional properties. *J. Database Manag.*, 22(2):1–31.

ClimFractal Analyser: um ambiente de análise de séries temporais climáticas baseado em *workflows**

Santiago A. Nunes[1], José E. M. Colabardini[1], Priscila P. Coltri[3], Ana M. H. de Ávila[3], Luciana A. S. Romani[1,2], Caetano Traina Junior[1], Agma J. M. Traina[1], Elaine P. M. Sousa[1]

[1]Departamento de Ciência de Computação - ICMC/USP - São Carlos - SP - Brasil
santiago@icmc.usp.br, dudu@grad.icmc.usp.br
{caetano, agma, parros}@icmc.usp.br

[2]Emprapa Informática Agropecuária - Campinas - Brasil
luciana@cnptia.embrapa.br

[3]Cepagri - Universidade Estadual de Campinas - Campinas - Brasil
{pcoltri, avila}@cpa.unicamp.br

Abstract. *The climate changes of last ages shows that the analysis of climate data is a fundamental task, because these changes impact both social and economical. In this context, this work presents a tool that allows to perform several types of analysis considering the behavior variation of climate data. Also, this tool can use climate model data in a dynamic environment based on workflows. In this way, our tool helps the specialist to make decisions.*

Resumo. *As mudanças do comportamento do clima das últimas décadas demonstram que a análise de dados climatológicos é de fundamental importância devido ao impacto social e econômico que implicam. Nesse contexto, esse trabalho apresenta uma ferramenta que permite realizar diversos tipos de análises considerando a variação do comportamento desses dados assim como a utilização de dados provenientes de modelos climáticos, em um ambiente dinâmico baseado em workflows, de modo a auxiliar o especialista do domínio em sua tarefa de tomada de decisão.*

1. Introdução

O aumento da frequência de eventos meteorológicos extremos, juntamente com o aumento da temperatura e a mudança na distribuição de chuvas, indicam mudanças significativas no clima global. Nesse cenário, pesquisas que envolvam o clima e a análise de dados meteorológicos são de fundamental importância devido ao impacto social e econômico que essas mudanças implicam. Inicialmente, as pesquisas climatológicas visavam entender os efeitos do clima sobre o comportamento da sociedade, por exemplo, seus hábitos, distribuição geográfica e atividades como agricultura. Com o crescimento da população, as pesquisas intensificaram-se [Ayoade 1996] e atualmente, grandes conjuntos de dados meteorológicos provenientes de diferentes fontes, como estações meteorológicas, satélites, modelos climáticos, dentre outros, são analisados pelos especialistas a fim de entender eventos extremos e anomalias climáticas.

*Os autores agradecem o apoio financeiro das seguintes agências de fomento à pesquisa do Brasil: CNPq, Instituto Microsoft Research - Fapesp, RUSP.

Atualmente, o avanço da tecnologia utilizada na obtenção de dados meteorológicos resultou em um crescimento desses conjuntos de dados, dificultando assim, sua análise manual ou semiautomática. Desse modo, ferramentas computacionais tornaram-se necessárias para auxiliar os meteorologistas a analisar o clima presente e estimar o comportamento no futuro. No entanto, de acordo com os especialistas do domínio, há uma carência de ferramentas que integrem tarefas diversas do processo de análise, como seleção de dados, visualização, análise de comportamento (principalmente considerando múltiplas variáveis climáticas), mineração de dados, análise de similaridade, entre outras. Neste cenário, visando suprir essa carência de ferramentas de análise de dados meteorológicos, algumas delas vêm sendo propostas e desenvolvidas como é o caso do sistema *SatImage Explorer*[Chino et al. 2011], cuja finalidade é auxiliar a análise de imagens de satélites, utilizando técnicas de detecção de agrupamentos sobre séries temporais extraídas de imagens de satélites.

Nesse contexto, este trabalho propõe um ambiente dinâmico baseado em *workflows* para a análise de múltiplas séries temporais climáticas, denominado *ClimFractal Analyser*. Essa ferramenta utiliza técnicas da teoria dos fractais e medidas estatísticas básicas como média e desvio padrão, permitindo que o especialista determine dinamicamente, através da parametrização de um *workflow*, diversos cenários de análise de séries temporais climáticas sem a necessidade de prender-se a um único tipo de análise.

2. Técnicas e Trabalhos Correlatos

A manipulação de múltiplas séries temporais como *data streams* multidimensionais permite que técnicas utilizadas para mineração de *data streams* sejam utilizadas em séries temporais. Uma *data stream* pode ser definida como uma sequência continua de itens de dados ou eventos, ordenadas e potencialmente infinita e, em geral, o foco de análise está em um intervalo de eventos e não na *stream* inteira. Desse modo, o modelo baseado em janelas permite limitar o escopo da sequência de eventos a serem tratados.

Neste trabalho, são utilizadas técnicas baseadas na teoria dos fractais para realizar a análise do comportamento de *data streams* multidimensionais. Um fractal é definido pela propriedade de auto-similaridade, isto é, objetos fractais apresentam as mesmas características em diferentes variações de escala e tamanho. Desse modo, partes de um fractal são similares, exata ou estatisticamente, ao fractal como um todo[Schroeder 1991].

Na teoria dos fractais, existem conceitos relevantes para a tarefa de mineração de dados, dentre eles, pode-se citar a dimensão fractal, que provê uma estimativa da dimensionalidade intrínseca de um conjuntos de dados. A dimensão intrínseca fornece a dimensionalidade do objeto representado pelos dados, independente da dimensão do espaço em que está inserido. A dimensão fractal é uma estimativa do comportamento não-uniforme do conjunto de dados e indica a existência ou não de correlações entre os atributos dos eventos que compõem o conjunto de dados [Faloutsos and Kamel 1994, Traina Jr. et al. 2005].

A dimensão fractal de objetos estatisticamente auto-similares, como é o caso de grande parte dos conjuntos de dados reais, pode ser determinada pela Dimensão Fractal de Correlação D_2, podendo ser utilizada para estimar a dimensão intrínseca de conjuntos de dados reais com um custo computacional viável. A análise baseada na dimensão intrínseca pode ser estendida de modo a detectar mudanças de comportamento em *data*

streams evolutivas. Essa análise é realizada por meio da medição contínua da dimensão fractal de uma *data stream* ao longo do tempo. Assim, variações significativas nas medidas sucessivas de D_2 podem indicar mudanças nas características intrínsecas dos dados. Um algoritmo eficiente para realizar a análise continuada da dimensão fractal de um conjunto de dados é o *SID-meter*, apresentado em [Sousa et al. 2007]. Trata-se de um algoritmo baseado em janelas deslizantes definidas sobre o conjunto de dados, permitindo assim delimitar o escopo de análise da *data streams* de modo a realizar o cálculo continuado da dimensão fractal.

Outro aspecto fundamental em Meteorologia é a previsão do tempo e do clima. Nesse contexto, diversos modelos climáticos foram criados visando suprir essa demanda. Um dos modelos que vêm sendo muito utilizado no Brasil é o ETA, instalado no Centro de Previsão de Tempo e Estudos Climáticos (CPTEC) em 1996[Chou 1996]. Este modelo é utilizado para analisar a variabilidade em várias escalas, desde mudanças nas médias anuais até ciclos diários. Nesse contexto, mecanismos que permitam analisar a conformidade entre os resultados gerados pelo modelo climático com séries temporais climatológicas reais são fundamentais para ajustar e melhorar os modelos climáticos assim como para validar os resultados obtidos por meio da análise dos resultados dos modelos. O método proposto em [Willmott et al. 1985] fornece um índice d, que reflete o grau de concordância entre as séries reais e de dados gerados por modelos climáticos. Esse índice avalia a exatidão e a divergência dos valores simulados em relação aos valores reais, variando de 0 (total discordância) a 1 (total concordância). Utilizando esse índice é possível inferir o quão efetivo é a utilização de séries temporais geradas por modelos climáticos em uma análise específica, além de permitir verificar qual modelo climático é o mais adequado a um determinado conjunto de dados reais.

3. A ferramenta *ClimFractal Analyser*

A ferramenta *ClimFractal Analyser*, desenvolvida com o objetivo de integrar diversos tipos de análise sobre dados meteorológicos, permite que o usuário configure o fluxo da análise por meio da parametrização de um *workflow*. A principio, a ferramenta permitia que um único tipo de análise fosse realizado. Porém, com o decorrer do projeto, a necessidade de outros tipos de análises surgiram, de modo que um ambiente dinâmico de análise tornou-se um requisito para o aprimoramento da ferramenta. Nesse contexto, tendo em vista a linguagem utilizada no desenvolvimento da ferramenta (C++) e a complexidade das análises que seriam realizadas, um novo ambiente baseado em *workflows* foi desenvolvido. Permitindo uma maior gama de análises sem se prender a nenhum fluxo de análise específico.

Ambientes baseados em *workflows* descrevem uma série de construções individuais que são incorporadas em um cenário com recursos já existentes, essas construções são decorrências de necessidades de uma modelagem real. Além disso, é possível que novas funcionalidades sejam incorporadas sem obrigar uma abordagem de implementação específica[Russell et al. 2006], facilitando assim, a tarefa de análise do especialista do domínio. Nesse contexto, diversas funcionalidades foram implementadas em componentes individuais na ferramenta *ClimFractal Analyser* e são descritas com maiores detalhes nas subseções seguintes.

A interface principal da ferramenta é apresentada na Figura 1, onde é possível vi-

sualizar como as funcionalidades individuais são combinadas de modo a realizar uma análise específica. Essa análise é definida em um *workflow*, construído por meio de uma interface onde o usuário arrasta os componentes desejados para o ambiente de parametrização, localizado do lado direito da interface. Esses componentes interagem por meio da definição dos fluxos de dados representados por setas entre os componentes.

3.1. Fonte de Dados

Este sistema foi desenvolvido visando manipular diversas fontes de dados. Desse modo, as fontes de dados utilizadas são parametrizáveis de duas maneiras: utilizando conexões com banco de dados, de modo que seja possível utilizar diversos bancos de dados simultaneamente e também por meio da utilização de arquivos no formato CSV (*comma-separated values*) contendo séries climáticas quaisquer.

Um ponto relevante em sistemas de previsão climática é que as medições de dados meteorológicos dependem de dispositivos analógicos, como sensores de temperatura, umidade e radiação solar, para prover a informação física a respeito das condições climáticas em dado momento. Esses dispositivos não são totalmente confiáveis, resultando esporadicamente na produção de medidas errôneas. Nesse contexto, por meio da utilização do método proposto em [Romani et al. 2003], o sistema permite realizar uma estimativa de dados faltantes, por meio da utilização de dados provenientes de estações próximas àquela que apresenta medições errôneas.

3.2. Filtros

Sobre as fontes de dados parametrizadas no *workflow* é possível aplicar filtros, permitindo que o especialista do domínio selecione regiões de interesse assim como intervalos de tempos de interesse.

3.3. Ferramentas de Análises Básicas

Sobre os conjuntos de dados parametrizados no *workflow*, o sistema possibilita que sejam realizados diversos tipos de análises explicadas com maiores detalhes a seguir:

- Análise do comportamento intrínseco do conjunto de dados: Por meio da implementação do algoritmo *SID-meter* em um componente, o sistema permite realizar múltiplas análises simultaneamente para diversas abordagens.
- Cálculo da média e do desvio padrão: A análise do comportamento intrínseco pode destacar o acontecimento de eventos extremos no conjunto de dados[Nunes et al. 2011]. Desse modo, a variação da média e do desvio padrão das variáveis que compõe o conjunto de dados pode indicar quais variáveis são responsáveis pelas mudanças de comportamento do conjunto de dados.
- Análise da conformidade entre séries temporais: Por meio da utilização do índice d de Willmott, é possível realizar uma análise sobre o grau de conformidade entre os conjuntos de dados reais e o conjunto de dados provenientes de modelos climáticos.

3.4. Visualização dos resultados

Os resultados obtidos na análise dos conjuntos de dados podem ser exibidos utilizando gráficos de linhas que permitem visualizar os resultados do cálculo da variação do comportamento do conjunto de dados realizados pelo *SID-meter*, assim como o cálculo da

variação índice d de Willmott e também a variação da média e desvio padrão de variáveis individuais.

Também é possível visualizar a localização geográfica do conjunto de dados. Esse tipo de saída permite que usuário selecione o tipo de API que será utilizada na exibição (GoogleMaps[1] ou Bing[2]) além de exibir informações individuais de cada estação meteorológica utilizada na análise.

4. Experimentos

Utilizando a ferramenta *ClimFractal Analyser*, o fluxo apresentado na Figura 1 foi construído. Nesse experimento, utilizou-se dados provenientes de uma estação meteorológica no estado de São Paulo, utilizando as variáveis climatológicas de temperatura média e precipitação no intervalo de 1961 a 1990. Utilizando a latitude e longitude da estação meteorológica, realizou-se uma busca pelos dados provenientes do modelo climático ETA para o mesmo ponto. Sobre esse conjunto de dados, realizou-se o cálculo continuado da dimensão fractal assim como o cálculo continuado do índice d de Willmott.

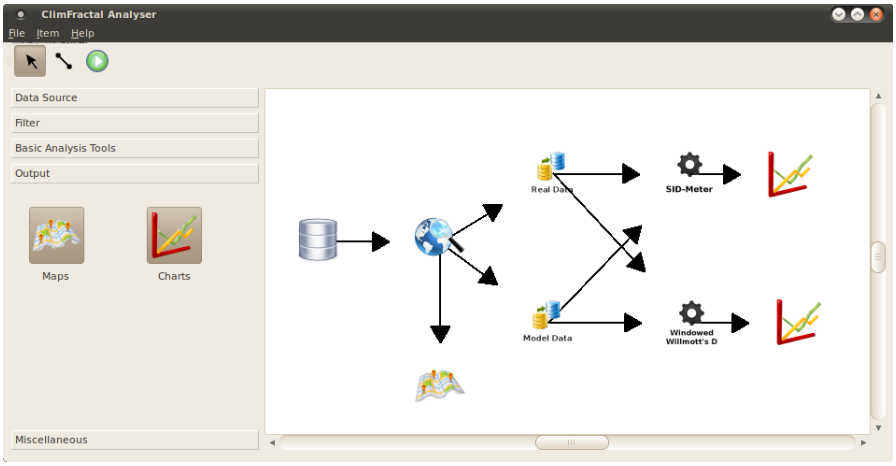


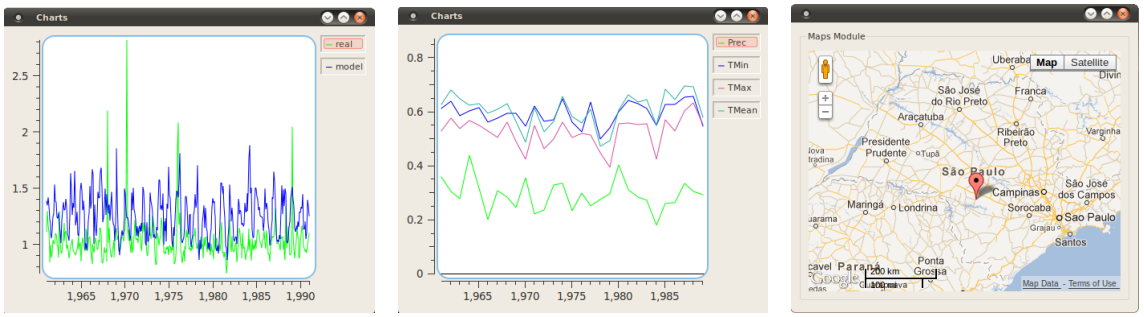
Figura 1. Interface da ferramenta *ClimFractal Analyser* com a definição de um fluxograma para análise de uma estação meteorológica.

Como resultado, tem-se a variação da dimensão fractal, apresentado na Figura 2(a), assim como a variação do índice d de Willmott, apresentado na Figura 2(b). Além disso, é possível visualizar a posição geográfica da estação meteorológica, como apresentado na Figura 2(c).

5. Conclusão

A ferramenta *ClimFractal Analyser* permite que diferentes tipos de análises sejam realizadas sobre diversos conjuntos de dados, em um ambiente dinâmico baseado em *workflows*, auxiliando o especialista do domínio em sua tarefa de descoberta de conhecimento. Devido a sua arquitetura baseada em módulos individuais, é possível estender essa ferramenta adicionando novas funcionalidades de forma simples.

[1]http://code.google.com/intl/pt-BR/apis/maps/index.html
[2]http://www.bing.com/maps/



(a) Variação de D_2.

(b) Variação do índice d de Willmott.

(c) Localização geográfica da estação meteorológica.

Figura 2. Análises realizadas pelo *ClimFractal Analyser*.

Como apresentado na seção anterior, é possível realizar experimentos complexos de forma dinâmica sem se prender a uma análise específica, empregando simultaneamente diversos algoritmos de modo a melhorar a compreensão do comportamento do conjunto de dados.

Referências

Ayoade, J. O. (1996). *Introdução à climatologia para os trópicos*. ed. Brestrand Brasil, Rio de Janeiro.

Chino, D. Y. T., Daniel, Y. T. C., Bruno, F. A., Luciana, A. S. R., Elaine, P. M. S., and Agma, J. M. T. (2011). Satimagexplorer: tornando a mineração de dados de sensores orbitais mais sensível. In *Brazilian Symposium on Databases, Florianópolis - SC*, volume 2, pages 25–30.

Chou, S. (1996). Modelo regional eta. climanálise. Technical report, Instituto Nacional de Pesquisas Espaciais.

Faloutsos, C. and Kamel, I. (1994). Beyond uniformity and independence: Analysis of r-trees using the concept of fractal dimension. In *Proceedings of the Thirteenth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, May 24-26, 1994, Minneapolis, Minnesota*, pages 4–13. ACM Press.

Nunes, S. A., Romani, L. A. S., de Ávila, A. M. H., Jr., C. T., de Sousa, E. P. M., and Traina, A. J. M. (2011). Fractal-based analysis to identify trend changes in multiple climate time series. *JIDM*, 2(1):51–58.

Romani, L. A. S., Santos, E. H., Evangelista, S. R. M., Assad, E. D., and Pinto, H. (2003). Utilização de estações vizinhas para estimativa de temperatura e precipitação usando o inverso do quadrado da distância. In *Anais do XIII Congresso Brasileiro de Agrometeorologia, Santa Maria - RS*, pages 717–718.

Russell, N., Arthur, van der Aalst, W. M. P., and Mulyar, N. (2006). Workflow Control-Flow Patterns: A Revised View. Technical report, BPMcenter.org.

Schroeder, M. (1991). *Fractals, Chaos, Power Laws: Minutes From an Infinite Paradise*. W. H. Freeman.

Sousa, E. P. M., Traina, A. J. M., Traina, C., and Faloutsos, C. (2007). Measuring evolving data streams' behavior through their intrinsic dimension. *New Gen. Comput.*, 25(1):33–60.

Traina Jr., C., Sousa, E. P. M., and Traina, A. J. M. (2005). *Using Fractals in Data Mining*, volume 1, page 30. Wiley/IEEE Press.

Willmott, C., Davis, R., Feddema, J., Klink, K., Legates, D., Rowe, C., Ackleson, S., and O'Donnell, J. (1985). Statistics for the evaluation and comparison of models. *JOURNAL OF GEOPHYSICAL RESEARCH*, 90:8995–9005.

MyGFT: um Módulo de Integração entre MySQL e Google Fusion Tables

Alexandre Savaris[1,2], Carmem Satie Hara[1], Aldo von Wangenheim[1,2]

[1]Universidade Federal do Paraná (UFPR) – Departamento de Informática
Caixa Postal 19.081 – 81.531-980 – Curitiba – PR – Brasil

[2]INCoD – Instituto Nacional para Convergência Digital
Universidade Federal de Santa Catarina (UFSC) – Departamento de Informática e Estatística – Sala 320 – 88.040-970 – Florianópolis – SC – Brasil

{asavaris,carmem}@inf.ufpr.br, awangenh@inf.ufsc.br

Abstract. *This work presents MyGFT, a storage engine for integrating MySQL DBMS and Google Fusion Tables, a cloud-based data management service. The module is described in terms of architecture, its integration to MySQL and how its use can be a viable alternative to data retrieval for controlled and structured vocabularies used in healthcare applications.*

Resumo. *Este trabalho apresenta o MyGFT[1], um storage engine para a integração do SGBD MySQL ao serviço de gerenciamento de dados em nuvem Google Fusion Tables. O módulo é descrito em termos de arquitetura, integração ao MySQL e de como sua utilização pode se tornar uma alternativa viável à recuperação de dados pertencentes a vocabulários controlados e estruturados utilizados em aplicações na área da saúde.*

1. Introdução

A extensão SQL/MED (*Management of External Data*), definida em meados de 2000, passou a integrar o padrão SQL com o objetivo de estabelecer uma metodologia de acesso a fontes de dados externas às instâncias relacionais. Pela sua utilização, é possível complementar os dados normalizados disponíveis em instâncias de bancos de dados relacionais com dados provenientes de origens diversas. Esse acesso a fontes de dados heterogêneas provê uma interface unificada (baseada na linguagem SQL) que permite o estabelecimento de relações diretas entre os conjuntos disponíveis [Melton 2002].

A gravação de dados externos em arquivos ou mesmo em bancos de dados com arquiteturas diversas é uma prática conhecida. Como estratégia alternativa, serviços orientados a dados em nuvem (DaaS – *Data as a Service*) têm se apresentado como uma opção viável ao armazenamento convencional [Zhou 2010], [Dikaiakos 2009]. Além de tornarem o acesso aos dados ubíquo, esses serviços podem auxiliar na redução da redundância de dados entre diferentes sistemas de informação. Neste cenário, consideram-se diferentes sistemas clientes acessando os mesmos serviços em nuvem via APIs oferecidas pelos próprios serviços.

[1] Este trabalho é parcialmente financiado pela Fundação Araucária projeto 22.741 e CNPq processo 484366/2011-4.

Visando prover acesso uniforme a dados armazenados localmente e em nuvem, este trabalho apresenta o MyGFT – um *storage engine* que objetiva a utilização do serviço de gerenciamento de dados em nuvem Google Fusion Tables (GFT) através do SGBD MySQL. O *storage engine* é desenvolvido como uma extensão modular do MySQL e segue a modalidade *pass-through* do padrão SQL/MED. Os dados armazenados no GFT são compartilhados por diversas aplicações e acessados da mesma forma que os dados armazenados localmente. Assim, para sistemas baseados em MySQL, o uso do MyGFT torna o acesso ao GFT transparente à aplicação. Em outras palavras, é possível escrever consultas SQL que envolvam tanto tabelas locais quanto tabelas no GFT, ficando a cargo do MyGFT o acesso a tabelas remotas. Esse acesso envolve a construção de uma requisição HTTP, o envio dessa requisição ao GFT e a recepção/tratamento (*parsing*) da resposta. Sem o MyGFT essas tarefas teriam de ser realizadas pela aplicação, bem como o processamento de junções com os dados armazenados localmente.

O trabalho é organizado como segue. Na seção dois são apresentados detalhes técnicos sobre o GFT; a seção três descreve o processo de desenvolvimento do MyGFT, partindo da sua integração com a arquitetura modular do MySQL até o processo de utilização durante a criação de tabelas; a seção quatro apresenta e discute um possível cenário de uso para o módulo desenvolvido; a seção cinco relaciona trabalhos correlatos, e a seção seis conclui o trabalho com as primeiras impressões sobre a utilização do módulo e com possíveis trabalhos futuros.

2. Google Fusion Tables

Disponibilizado em junho de 2009, o Google Fusion Tables (GFT) é um serviço de gerenciamento de dados em nuvem que objetiva facilitar o compartilhamento de conjuntos de dados e a execução de atividades colaborativas sobre esses conjuntos em uma arquitetura *web* [Gonzalez 2010a]. O serviço possibilita a criação de tabelas de dados pela execução de comandos em uma interface *web* ou pela importação de arquivos nos formatos CSV (*Comma Separated Values*), KML (*Keyhole Markup Language*) ou planilhas, limitados a um tamanho máximo de 100MB. Uma vez criadas, essas tabelas de dados podem ser definidas como públicas, privadas ou compartilhadas entre usuários chamados *colaboradores*; além disso, seu conteúdo pode ser relacionado ao conteúdo de outras tabelas via equijunções, sendo possível também a construção de visões para a integração de diferentes tabelas visando um acesso unificado.

Estruturalmente, o GFT é organizado sobre uma pilha de serviços de armazenamento, com destaque para a estrutura utilizada na persistência de pares chave/valor e para a biblioteca de primitivas utilizada na criação de índices secundários, gerenciamento de transações e replicação [Gonzalez 2010b]. O armazenamento efetivo dos dados do GFT é feito em estruturas do tipo *Bigtable*, implementadas como mapas multidimensionais ordenados altamente escaláveis. Essas estruturas são utilizadas também para o armazenamento dos esquemas das tabelas, índices, *logs* de transação e comentários feitos pelos usuários com acesso às tabelas. A biblioteca *Megastore*, por sua vez, é responsável pela indexação secundária dos atributos das tabelas, pela implementação e gerenciamento de transações ACID e pela replicação de dados, esquemas, índices, *logs* e comentários em diferentes servidores.

A API[2] do GFT é organizada de forma a permitir que diferentes aplicações clientes possam usufruir do serviço de armazenamento, utilizando-se de um conjunto bem definido de operações DDL (criação e exclusão de tabelas) e DML (seleção, projeção, agrupamento e agregação, inserção, atualização e exclusão de dados). As operações a serem executadas são enviadas pelas aplicações clientes via HTTP/RPC para o GFT, onde são interceptadas e avaliadas pelo módulo despachante. Esse módulo converte a requisição recebida para uma representação interna do serviço (consulta), e a encaminha para o módulo de otimização, responsável pela geração do respectivo plano de execução. Esse plano de execução é enviado ao processador de consulta, módulo responsável pela execução do plano, pelo recebimento dos resultados e pelo encaminhamento desses resultados aos módulos de nível superior.

3. O *storage engine* MyGFT

O SGBD MySQL é caracterizado por uma arquitetura modular, extensível pelo desenvolvimento de *plugins* responsáveis por encapsular funcionalidades como busca textual e autenticação [Golubchik e Hutchings 2010]. Especificamente com relação ao armazenamento de dados, é possível definir e implementar diferentes formatos e suas respectivas formas de acesso pelo atendimento às especificações da *Storage Engine API*[3]. O *storage engine* MyGFT é construído de acordo com os preceitos dessa API, permitindo sua integração à arquitetura do MySQL conforme exibido na figura 1.

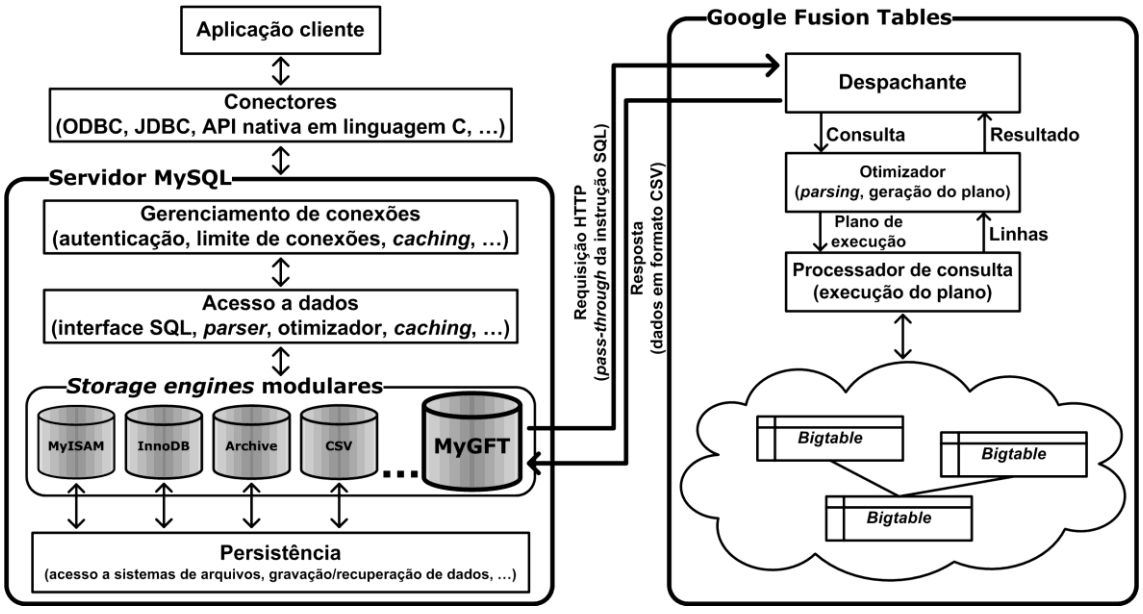


Figura 1. O *storage engine* MyGFT – integração à arquitetura MySQL e acesso ao GFT (adaptação).

Visando atender ao cenário de uso descrito na seção quatro, o MyGFT foi projetado como um *storage engine* somente de leitura; essa restrição de funcionalidade é possível graças à organização da *Storage Engine API*, que mapeia instruções DDL e DML para funções em linguagem C++ nas quais as ações são efetivamente implementadas.

[2] https://developers.google.com/fusiontables/
[3] http://dev.mysql.com/doc/refman/5.6/en/pluggable-storage-overview.html

Como exemplo, uma instrução SELECT é mapeada para um conjunto de funções (`open()`, `rnd_init()`, `rnd_next()`, `rnd_end()`, `close()`) invocadas pelo servidor na ordem definida pela API.

Com o módulo devidamente instalado, é possível criar tabelas e usá-las para recuperar dados armazenados no GFT. A criação dessas tabelas requer a especificação de um padrão de codificação de caracteres para os dados recuperados, bem como a especificação do identificador único da tabela no GFT:

```
mysql> install plugin MYGFT soname 'ha_mygft.so';

mysql> CREATE TABLE cid10(codigo_subcategoria text,
         nome_subcategoria text) DEFAULT CHARSET=utf8
         CONNECTION='4371448' engine=mygft;
```

No exemplo, o *default charset* para os dados recuperados é definido como UTF-8 (padrão do GFT), e o identificador único da tabela (4371448) corresponde ao identificador atribuído pelo GFT no momento da criação da tabela original. Além disso, a relação de campos deve ser equivalente à relação de colunas da tabela original.

Instruções de seleção (SELECTs) repassadas pelo servidor MySQL ao módulo MyGFT não são interpretadas pelo módulo. Graças à API SQL do GFT, é possível encaminhar essas instruções na forma de requisições HTTP diretamente ao serviço, provendo assim uma implementação para a modalidade *pass-through* prevista no padrão SQL/MED. A construção das requisições HTTP baseada na URL padrão para execução de consultas no GFT, bem como no identificador único da tabela no serviço, é feita com o auxílio das funções disponibilizadas pela biblioteca libcurl[4], que assume o papel de um cliente *web* integrado ao módulo. Essa abordagem simplifica a implementação do MyGFT, que repassa toda a responsabilidade pela manutenção dos dados pesquisados – incluindo a indexação – ao GFT.

Os dados encontrados pelo GFT após a execução das consultas são encaminhados ao MyGFT no formato CSV. O módulo, então, executa um *parsing* sobre esses dados convertendo-os no formato interno utilizado pelo MySQL para a representação de campos e registros, repassando-os sem seguida à camada de acesso a dados do servidor. Deve-se observar que os dados recuperados do GFT não são replicados na base local, mas apenas transformados no formato interno do SGBD para o processamento das consultas. Assim, os registros recuperados podem ser usados de forma integrada a registros armazenados em tabelas locais (via junções ou subconsultas), como no exemplo a seguir: uma tabela local (*laudo_exame*) é relacionada à tabela *cid10* armazenada no GFT, objetivando recuperar o identificador único do laudo e o termo relacionado ao vocabulário controlado/estruturado (vide seção 4).

```
mysql> SELECT le.id, c.nome_subcategoria FROM laudo_exame le, cid10 c
        WHERE le.codigo_subcategoria = c.codigo_subcategoria;
+----+-------------------------------------------------------+
| id | nome_subcategoria                                     |
+----+-------------------------------------------------------+
|  1 | Cólera devida a Vibrio cholerae 01, biótipo cholerae  |
+----+-------------------------------------------------------+
```

[4] http://curl.haxx.se/libcurl/

4. Cenário de uso – acesso a vocabulários controlados

Sistemas de informação desenvolvidos para a área da saúde costumam utilizar um ou mais vocabulários controlados/estruturados. Esses vocabulários são compostos por conjuntos de termos normalizados que objetivam padronizar a nomenclatura usada na área de forma a facilitar a indexação de conteúdo e diminuir a utilização de texto livre. Vocabulários como DeCS (Descritores em Ciências da Saúde)[5] e CID-10 (Classificação Estatística Internacional de Doenças e Problemas Relacionados com a Saúde)[6] são exemplos desses vocabulários; sua estruturação hierárquica permite representar não apenas doenças, mas também termos pertinentes a áreas como anatomia, compostos químicos, equipamentos, dentre outras.

Para que o conteúdo desses vocabulários seja disponibilizado às aplicações, é uma prática conhecida que suas estruturas hierárquicas sejam modeladas de maneira *ad hoc* em bancos de dados relacionais, resultando em diversos esquemas com particularidades específicas a cada aplicação. Essa multiplicidade de esquemas pode influenciar negativamente o processo de integração de dados entre sistemas (operação comum na área da saúde), bem como a busca de dados executada entre sistemas.

A utilização do *storage engine* MyGFT provê uma alternativa viável à criação de repositórios individuais (em cada sistema) para o armazenamento dos vocabulários controlados/estruturados. Sua característica somente de leitura permite que diferentes sistemas possam acessar o conteúdo de tabelas/visões criadas diretamente no GFT, centralizando o acesso a conjuntos de dados comuns e que, tais como esses vocabulários, não recebem atualizações constantes; de forma complementar, a restrição de acesso imposta ao módulo garante a integridade das informações disponibilizadas, evitando que diferentes aplicações atualizem de forma indiscriminada o seu conteúdo e comprometam a semântica inerente a dados utilizados previamente. Outro aspecto favorável à utilização do módulo é a garantia de que, em caso de atualização de conteúdo, o mesmo seja disponibilizado imediatamente a todas as instâncias locais em MySQL relacionadas às tabelas no GFT; com isso, garante-se que todas as aplicações que compartilham os dados dos vocabulários passem a usufruir de uma visão atualizada, provendo uma integração de dados consistente.

5. Trabalhos relacionados

Módulos customizados têm sido usados como forma de integração entre o modelo relacional e outros modelos de armazenamento, visando o aproveitamento das suas melhores características. O trabalho de [Ribas 2010] apresenta um módulo de armazenamento para a integração do MySQL com Tabelas de Espalhamento Distribuídas (DHT); diferentemente deste trabalho (que foca no acesso a dados centralizados utilizando clientes distribuídos), os autores visam o desenvolvimento de um sistema caracterizado pela escalabilidade, descentralização, tolerância a falhas e facilidade de uso, no qual os dados são distribuídos. O trabalho de [Atwood 2007], por sua vez, objetiva o armazenamento de grandes volumes de dados em nuvem, visando escalabilidade. No presente trabalho,

[5] http://decs.bvs.br/P/decsweb2012.htm
[6] http://www.datasus.gov.br/cid10/v2008/cid10.htm

busca-se a centralização e o compartilhamento de dados pouco mutáveis, com volumes bem definidos, cuja escalabilidade não é um fator determinante.

6. Conclusões e trabalhos futuros

Este trabalho apresentou o *storage engine* MyGFT, um módulo desenvolvido com o objetivo de integrar o SGBD MySQL ao serviço de gerenciamento de dados em nuvem Google Fusion Tables. Essa integração visa permitir a recuperação de conjuntos de dados armazenados em um repositório centralizado, de forma que os mesmos não precisem estar fisicamente replicados em diferentes sistemas de informação, estruturados em esquemas definidos de maneira *ad hoc*.

Os testes prévios executados com o módulo, em um cenário de uso envolvendo os vocabulários controlados/estruturados DeCS e CID-10, atestam a possibilidade de recuperação de termos desses vocabulários via MyGFT pela execução de consultas cujos predicados intersectem os conjuntos de predicados suportados pelo GFT e pelo MySQL. O desempenho de busca, conforme esperado, é inferior ao desempenho de acesso a dados locais; isso é justificável pelo fato do acesso ao GFT via MyGFT envolver, além da execução da consulta em si e do *parsing* sobre os dados encontrados, a construção de uma requisição HTTP, o envio dessa requisição ao GFT, a construção de uma resposta HTTP e o envio do conjunto de dados resultante ao MyGFT. Otimizações deste processo podem ser relacionadas como trabalhos futuros, englobando testes mais abrangentes envolvendo um maior número de predicados e a implementação de *caches* locais para a minimização do volume de dados trafegado.

Referências

Atwood, M. (2007) “A Storage Engine for Amazon S3”. Em: MySQL Conference & Expo 2007.

Dikaiakos, M. D., et al. (2009) “Cloud Computing: Distributed Internet Computing for IT and Scientific Research”, Em: IEEE Internet Computing, v. 13(5), p. 10-13.

Golubchik, S. e Hutchings, A. (2010), MySQL 5.1 Plugin Development, Packt Publishing Ltd.

Gonzalez, H., et al. (2010a) “Google Fusion Tables: Web-Centered Data Management and Collaboration”, Em: Proceedings of the 2010 International Conference on Management of Data, p. 1061-1066.

Gonzalez, H., et al. (2010b) “Google Fusion Tables: Data Management, Integration and Collaboration in the Cloud”, Em: Proceedings of the 1st ACM Symposium on Cloud Computing, p. 175-180.

Melton, J., et al. (2002) “SQL/MED – A Status Report”, Em: ACM SIGMOD Record, v. 31(3), p. 81-89.

Ribas, E. A., et al. (2010) “Um SGBD com Armazenamento Distribuído de Dados Baseado em DHT”, Em: Anais do XXV Simpósio Brasileiro de Banco de Dados.

Zhou, M., et al. (2010) “Services in the Cloud Computing Era: A Survey”, Em: Proceedings of the 4th International Universal Communication Symposium, p. 40-46.